

VŠB – Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky
Katedra aplikované matematiky

Metody rozložení oblasti urychlené multigridem v 2d

2d multigrid domain decomposition methods

Zadání diplomové práce

Student: **Bc. Sára Linhartová**
Studijní program: N2647 Informační a komunikační technologie
Studijní obor: 1103T031 Výpočetní matematika
Téma: **Metody rozložení oblasti urychlené multigridem v 2d**
2d multigrid domain decomposition methods
Jazyk vypracování: čeština

Zásady pro vypracování:

Řešení mnoha důležitých úloh např. aerodynamika letadla je možné pouze s využitím super-počítače. Řešení je založeno na rozložení výpočetní oblasti do podoblastí, kde se paralelně řeší menší tzv. lokální úloha. Tato řešení se pak korigují globálně tzv. hrubým problémem. V současnosti se umí řešit 3d úlohy o miliardách rovnic a neznámých. Výpočty narážejí na kvadratickou složitost Gaussovy eliminace, která se používá pro řešení lokálních úloh i hrubého problému. Cílem této práce je nahradit Gaussovu eliminaci multigridem, který má složitost pouze lineární. Algoritmus bude testován na diskretizaci 2d úlohy vedení tepla v heterogenním prostředí.

Seznam doporučené odborné literatury:

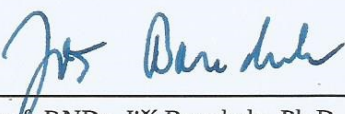
- [1] Toselli, Widlund - Domain decomposition methods. Springer, 2005
- [2] Lukáš, Bouchala, Vodstrčil, Malý - 2-dimensional primal domain decomposition theory in detail. Applications of Mathematics 60, pp. 265-283, 2015

Formální náležitosti a rozsah diplomové práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.


Vedoucí diplomové práce: **doc. Ing. Dalibor Lukáš, Ph.D.**

Datum zadání: 01.09.2018

Datum odevzdání: 30.04.2019

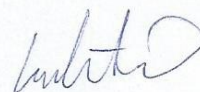

prof. RNDr. Jiří Bouchala, Ph.D.
vedoucí katedry




prof. Ing. Pavel Brandštetter, CSc.
děkan fakulty

Prohlašuji, že jsem tuto diplomovou práci vypracovala samostatně. Uvedla jsem všechny literární
prameny a publikace, ze kterých jsem čerpala.

V Ostravě 26. dubna 2019


.....

Souhlasím se zveřejněním této diplomové práce dle požadavků čl. 26, odst. 9 Studijního a zkušebního řádu pro studium v magisterských programech VŠB-TU Ostrava.

V Ostravě 28. dubna 2019

.....

Ráda bych na tomto místě poděkovala všem, kteří mi s prací pomohli, protože bez nich by nevznikla. Jmenovitě Doc. Ing. Daliborovi Lukášovi, Ph.D. za čas a trpělivost, kterou mi věnoval.

Abstrakt

Tato práce se zabývá optimalizací výpočetní náročnosti metody rozložení oblasti ve dvou dimenzích pomocí nahrazení Gaussovy eliminace s kvadratickou složitostí multigridem, jehož složitost je lineární. Na úvod se text věnuje metodě konečných prvků, která je nejpoužívanější metodou při numerickém řešení parciálních diferenciálních rovnic. Dále je popsána metoda rozložení oblasti, která umožňuje rozdělit problém na lokální úlohy a řešit je paralelně. Později text popisuje multigrid a algoritmus metody dvou sítí. Na závěr jsou pak srovnány výsledky pro metodu rozložení oblasti s využitím Gaussovy eliminace a metodu rozložení oblasti, kde je Gaussova eliminace nahrazena metodou dvou sítí.

Klíčová slova: metoda konečných prvků, metoda rozložení oblasti, multigrid

Abstract

This thesis is focused on two dimensional Domain decomposition method optimization by replacing Gauss elimination, with the quadratic complexity, by Multigrid whose complexity is linear. The introductory text deals with the finite element method, which is the most used method in numerical solution of partial differential equations. Next, the method of domain decomposition is described, which allows to divide the problem into local tasks and solve them parallel. Further text describes multigrid and the two-grid algorithm. Finally, there are compared two variants of domain decomposition method. The standart one, using Gauss elimination method, is compared to a novel one, where Gauss elimination method is replaced by two-grid method.

Key Words: finite element method, domain decomposition method, multigrid

Obsah

Seznam obrázků	8
Seznam tabulek	9
1 Úvod	10
2 Řešení eliptických PDR metodou konečných prvků ve 2D	11
2.1 Eliptické parciální diferenciální rovnice druhého řádu	11
2.2 Slabá formulace okrajových úloh	12
2.3 Metoda konečných prvků	15
3 Metoda rozložení oblasti ve 2D	19
3.1 Teoretický základ	19
3.2 Vlastní implementace	24
3.3 Numerické experimenty	29
4 Multigrid ve 2D	38
4.1 Teoretický základ	38
4.2 Vlastní implementace	42
5 Optimalizace metody rozložení oblasti pomocí metody dvou sítí ve 2D	43
5.1 Numerické experimenty	44
6 Závěr	46
Literatura	49

Seznam obrázků

1	Rozložení oblasti Ω na $N \times N = 9$ podoblastí s $n^V = 4$ hraničními uzly hrubé sítě x_i^V (vyznačeno černými kosočtverci); přerušovaná čára označuje $\partial\Omega$; silné plné čáry označují skeleton Γ rozložený na $M = 12$ vnitřních hran E_i s hraničními uzly $x_{i,j}^E$ (vyznačeno prázdnými kruhy); slabé plné čáry označují jemné dělení s $n = 81$ vnitřními uzly $x_{i,j}^I$ (vyznačeno menšími plnými kruhy).	20
2	Rozložení oblasti $\Omega = \langle 0, 1 \rangle \times \langle 0, 1 \rangle$ pro $N = 2$ a $\frac{H}{h} = 3$	26
3	Očíslování uzlů jedné podoblasti při tvorbě lokální matice A_{LOC}	27
4	Struktura lokální matice A_{LOC}	27
5	Očíslování uzlů oblasti Ω při tvorbě globální matice A	28
6	Struktura globální matice A	28
7	Diskretizace oblasti Ω	30
8	Číslo podmíněnosti $\kappa(\hat{\mathbf{S}}^{-1}\mathbf{S})$ pro $N = 2, 4, 6, \dots, 30$ a $\frac{H}{h} = 2, 3$ v logaritmickém měřítku	31
9	Perturbace matice \mathbf{A} pro $N = 2$, $\frac{H}{h} = 4$	32
10	Perturbace matice \mathbf{A} pro $N = 2$, $\frac{H}{h} = 16$	33
11	Perturbace matice \mathbf{A} pro $N = 2$, $\frac{H}{h} = 16$. Změna čísla podmíněnosti $\kappa(\hat{\mathbf{S}}^{-1}\mathbf{S})$ po přičtení náhodných hodnot k vlastním číslům matice \mathbf{A} (modrá) a původní číslo podmíněnosti (červená)	33
12	Perturbace matice \mathbf{A} pro $N = 4$, $\frac{H}{h} = 4$	34
13	Perturbace matice \mathbf{A} pro $N = 6$, $\frac{H}{h} = 4$	34
14	Perturbace matice \mathbf{A} pro $N = 8$, $\frac{H}{h} = 4$	35
15	Ilustrace báзовých funkcí hrubé sítě v jedné dimenzi; $\varphi_1(x)$ báзовá funkce uzlu 1 značena tečkovanou čárou; $\varphi_2(x)$ báзовá funkce uzlu 2 značena plnou čárou; $\varphi_3(x)$ báзовá funkce uzlu 3 značena přerušovanou čárou	40
16	Ilustrace báзовé funkce uzlu jemné sítě (černě) ve srovnání s báзовou funkcí původního uzlu 2 hrubé sítě (šedě) v jedné dimenzi	40
17	Ilustrace báзовé funkce uzlu 5 hrubé sítě ve dvou dimenzích	41
18	Ilustrace báзовé funkce uzlu jemné sítě (černě) ve srovnání s báзовou funkcí původního uzlu 5 hrubé sítě (šedě) ve dvou dimenzích	41

Seznam tabulek

1	Maximální odchylka čísla podmíněnosti $\kappa(\hat{\mathbf{S}}^{-1}\mathbf{S})$ po perturbaci matice \mathbf{A} pro různé parametry $\frac{H}{h}$	32
2	Maximální odchylka čísla podmíněnosti $\kappa(\hat{\mathbf{S}}^{-1}\mathbf{S})$ po perturbaci matice \mathbf{A} pro měnící se parametr N	34
3	Počet iterací předpodmíněné metody sdružených gradientů pro $\frac{H}{h} = 4$ a měnící se parametr N	37
4	Počet iterací předpodmíněné metody sdružených gradientů pro $\frac{H}{h} = 16$ a měnící se parametr N	37
5	Počet iterací předpodmíněné metody sdružených gradientů za využití metody dvou sítí pro $\frac{H}{h} = 4$ a měnící se parametr N	45
6	Počet iterací předpodmíněné metody sdružených gradientů za využití metody dvou sítí pro $\frac{H}{h} = 16$ a měnící se parametr N	45

1 Úvod

Řešení mnoha důležitých úloh (např. průběh napětí, deformace těles, proudění kapalin) je založeno na rozložení výpočetní oblasti do podoblastí, kde můžeme paralelně řešit menší tzv. lokální úlohy. V současné době umíme díky superpočítačům řešit úlohy o miliardách neznámých, nicméně výpočty narážejí na kvadratickou složitost Gaussovy eliminace, která se používá k řešení lokálních i globálního problému. Cílem této práce je nahradit Gaussovu eliminaci multigridem, který má složitost pouze lineární.

Úvodem je třeba položit základy, na kterých můžeme řešení cíle stavět. Nejdříve popíšeme úlohu eliptických parciálních diferenciálních rovnic druhého řádu. Poté navážeme metodou konečných prvků, pomocí níž umíme úlohu řešit. Metoda konečných prvků pro nás bude zdrojem soustavy lineárních rovnic $Au = b$ a vzhledem k tomu, že se v práci zabýváme výpočetní náročností řešení, jednou ze stěžejních informací bude podmíněnost matice soustavy \mathbf{A} .

Na metodě konečných prvků se pak zakládá metoda rozložení oblasti, jejíž cílem je najít paralelní symetrické pozitivně definitní (SPD) předpodmínění $\hat{\mathbf{A}}^{-1}$, které je cestou k lepší podmíněnosti soustavy. Metoda rozložení oblasti rozdělí problém na lokální úlohy, které je možné řešit paralelně. Tato práce má poskytnout pouze nástroj k paralelizaci, proto se jí hlouběji zabývat nebudeme. Na druhou stranu podrobně rozebereme algoritmus, který vede k optimalizované soustavě, podpoříme teoretické základy numerickými testy a budeme se věnovat i způsobu implementace metody rozložení oblasti.

Dále si představíme multigrid, respektive metodu dvou sítí, popíšeme její algoritmus, některé vlastnosti a podíváme se i na její implementaci.

Stěžejní vlastností multigridu je jeho lineární složitost, které využijeme v závěrečné kapitole. Tato část práce se bude zabývat nahrazením Gaussovy eliminace multigridem a analýzou vlivu multigridu na rychlost konvergence při hledání řešení diskretizační úlohy.

2 Řešení eliptických PDR metodou konečných prvků ve 2D

2.1 Eliptické parciální diferenciální rovnice druhého řádu

Následující text čerpá z [4].

Velké množství důležitých matematických úloh je popsáno pomocí parciálních diferenciálních rovnic (PDR). V případě reprezentační úlohy této práce se jedná o dvoudimenzionální úlohu vedení tepla.

Nejprve si připomeňme definici parciálních diferenciálních rovnic druhého řádu a jejich klasického řešení:

Definice 1 *Parciální diferenciální rovnici (PDR) rozumíme rovnici, která kromě neznámé funkce (alespoň dvou proměnných) obsahuje také její (parciální) derivace.*

Definice 2 *Nechť $F : \mathbb{R}^k \times \mathbb{R}^{k-1} \times \dots \times \mathbb{R}^n \times \mathbb{R} \times U \rightarrow \mathbb{R}$ je známá funkce. Rovnice*

$$F(D^k u(x), D^{k-1} u(x), \dots, Du(x), u(x), x) = 0 \quad (1)$$

pro $x \in U$, kde $u : U \rightarrow \mathbb{R}$ je neznámá funkce, se nazývá parciální diferenciální rovnice k -tého řádu.

Definice 3 *Klasickým řešením PDR k -tého řádu se rozumí funkce $u : U \rightarrow \mathbb{R}$ splňující (1) pro všechna $x \in U$ a má spojitě parciální derivace až do řádu k včetně, kdy píšeme $u \in C^k$, resp. $u \in C^k(U)$.*

Řešit parciální diferenciální rovnici analyticky je možné jen ve speciálních případech, což je problém, protože se jedná o ty nejpoužívanější rovnice ve vědě a inženýrství. Naštěstí díky dnešní pokročilé počítačové technice můžeme řešit složité úlohy PDR numericky.

Velké množství úloh spadá do nejjednodušší kategorie lineárních PDR 2. řádu, což jsou rovnice ve tvaru

$$\sum_{i=1}^n \sum_{j=1}^n a_{i,j}(x) \cdot \frac{\partial^2 u}{\partial x_i \partial x_j}(x) + \sum_{i=1}^n b_i(x) \cdot \frac{\partial u}{\partial x_i} + c \cdot u(x) = f(x).$$

I tato množina rovnic je stále poměrně náročná při řešení, proto se obvykle omezujeme na rovnice s konstantními koeficienty. My se budeme zabývat rovnicemi ve dvou dimenzích, kdy je tradiční zápis

$$A \frac{\partial^2 u}{\partial x^2} + B \frac{\partial^2 u}{\partial x \partial y} + C \frac{\partial^2 u}{\partial y^2} + D \frac{\partial u}{\partial x} + E \frac{\partial u}{\partial y} + F y = f.$$

Členy s první derivací lze „ukrýt“ vhodnou substitucí, proto jsou pro chování rovnice rozhodující členy s druhou derivací, tedy hodnoty konstant A , B , C . Podle nich se tyto rovnice dělí

na eliptické, parabolické a hyperbolické. Tyto základní typy naznačí, jak vypadá situace ve více proměnných. Rovnice převoditelné na tvar $\sum_{i=1}^n \frac{\partial^2 u}{\partial x_i^2} = f$ se nazývají eliptické.

Rozdíl mezi eliptickými a ostatními druhy PDR druhého řádu je vnímání proměnných. U eliptických rovnic jsou všechny proměnné rovnocenné, jde o proměnné prostorové. V případě parabolických a hyperbolických rovnic je jedna proměnná časová a ostatní jsou prostorové. Ve všech třech případech pracujeme na nějaké pevně dané oblasti $\Omega \subseteq \mathbb{R}^n$.

V případě eliptických rovnic, máme-li funkci u s prostorovými souřadnicemi x_1, \dots, x_n , zavádíme Laplaceův operátor

$$\Delta u = \sum_{i=1}^n \frac{\partial^2 u}{\partial x_i^2}.$$

U parciálních diferenciálních rovnic potřebujeme mít k určení jednoznačného řešení doplňující podmínky. V případě eliptických rovnic je typická „okrajová podmínka“, kdy jsou pevně zadány hodnoty pro u na hranici oblasti Ω .

2.2 Slabá formulace okrajových úloh

Následující text čerpá z [5].

Dříve, než se dostaneme k metodě konečných prvků (MKP), je třeba definovat slabou formulaci okrajové úlohy. Postup si popíšeme na následujícím příkladě:

Okrajový problém:

Nechť Ω je omezená oblast v \mathbb{R}^2 s lipschitzovskou hranicí $\partial\Omega$, $\partial\Omega = \Gamma_D \cup \Gamma_N$. Část Γ_D a Γ_N hranice oblasti Ω jsou tvořeny konečným počtem uzavřených oblouků, které mají společné nejvýše krajní body. Mějme dány funkce $\rho_{i,j} : \bar{\Omega} \rightarrow \mathbb{R}, i, j = 1, 2, f : \Omega \rightarrow \mathbb{R}, u_D : \Gamma_D \rightarrow \mathbb{R}$ a $g_N : \Gamma_N \rightarrow \mathbb{R}$. Hledáme funkci $u : \bar{\Omega} \rightarrow \mathbb{R}$, která řeší rovnici

$$-\sum_{i=1}^2 \frac{\partial}{\partial x_i} \left(\sum_{j=1}^2 \rho_{ij}(x) \frac{\partial u}{\partial x_j}(x) \right) = f(x) \quad \forall x \in \Omega \quad (2)$$

a splňuje okrajové podmínky

$$u(x) = u_D(x) \quad \forall x \in \Gamma_D \quad (3)$$

$$\sum_{i=1}^2 \left(\sum_{j=1}^2 \rho_{ij}(x) \frac{\partial u}{\partial x_j} \right) n_i(x) = g_N(x) \quad \forall x \in \Gamma_N \quad (4)$$

kde $n(x) = (n_1(x), n_2(x))$ je vektor vnější normály k $\partial\Omega$ v bodě x . Funkci $u \in C^2(\bar{\Omega})$, která splňuje (2), (3) a (4), nazýváme klasickým řešením problému.

Slabá formulace:

Protože klasické řešení nemusí vždy existovat, zavedeme pojem slabé řešení či slabá formulace úlohy. K jeho odvození potřebujeme znát znění Greenovy věty:

Věta 1 (Greenova)

i) Necht $1 < n \in \mathbb{N}$ a necht $\emptyset \neq \Omega \subset \mathbb{R}^n$ je oblast s lipschitzovskou hranicí. Pak pro každé $u, v \in W^{1,2}(\Omega)$ a $i \in \{1, 2, \dots, n\}$ platí

$$\int_{\Omega} u \frac{\partial v}{\partial x_i} dx = - \int_{\Omega} \frac{\partial u}{\partial x_i} v dx + \int_{\partial\Omega} T(u) T(v) \nu_i ds$$

($\nu_i = \nu_i(x)$... i -tá složka jednotkového vektoru vnější normály k $\partial\Omega$ v x).

ii) Buď $a, b \in \mathbb{R}$, $a < b$. Pak $W^{1,2}(a, b) \subset C(\langle a, b \rangle)$ a navíc

$$\forall u, v \in W^{1,2}(a, b) : \int_a^b uv' dx = - \int_a^b u'v dx + u(b)v(b) - u(a)v(a).$$

Důkaz Greenovy věty viz [?].

Nyní označme

$$V = \{v \in X; v|_{\Gamma_D} = 0\}, \quad (5)$$

kde $X = H^1(\Omega)$ je Sobolevův prostor tvořený funkcemi, které jsou splnou se svými zobecněnými derivacemi prvního řádu integrovatelné s druhou mocninou přes Ω , a kde rovnost $v|_{\Gamma_D} = 0$ chápeme ve smyslu stop.

Nejdříve předpokládejme, že funkce $b_{i,j}$, f , u_D a g jsou dostatečně hladké a klasické řešení u problému existuje. Vynásobme rovnici (2) libovolnou funkcí $v \in V$, integrujme získanou rovnost přes oblast Ω a aplikujme na integrál na jeho levé straně Greenovu větu. Dostaneme

$$\sum_{i=1}^2 \int_{\Omega} \left(\sum_{j=1}^2 \rho_{ij} \frac{\partial u}{\partial x_j} \right) \frac{\partial v}{\partial x_i} dx - \sum_{i=1}^2 \int_{\partial\Omega} \left(\sum_{j=1}^2 \rho_{ij} \frac{\partial u}{\partial x_j} \right) \nu_i dS = \int_{\Omega} f v dx.$$

Využijeme-li toho, že $\int_{\partial\Omega} = \int_{\Gamma_D} + \int_{\Gamma_N}$, klasické řešení u splňuje podmínku (4) a funkce v je nulová na Γ_D (ve smyslu stop), zjistíme, že

$$\int_{\Omega} \sum_{i=1}^2 \left(\sum_{j=1}^2 \rho_{ij} \frac{\partial u}{\partial x_j} \right) \frac{\partial v}{\partial x_i} dx = \int_{\Omega} f v dx + \int_{\Gamma_N} g_N v dS. \quad (6)$$

Všimněme si, že k tomu, aby byly integrály v (6) definovány (a konečné), nemusí být nutně $u \in C^2(\bar{\Omega})$ a funkce $\rho_{i,j}$, f a g_N nemusí být příliš hladké. Stačí, když bude $u \in H^1(\Omega)$ a

$$\rho_{i,j} \in L^\infty(\Omega), \quad f \in L^2(\Omega) \quad \text{a} \quad g_N \in L^2(\Gamma_N). \quad (7)$$

Pro $v, w \in H^1(\Omega)(=X)$ označme

$$\begin{aligned} a(w, v) &= \int_{\Omega} \sum_{i=1}^2 \left(\sum_{j=1}^2 \rho_{ij} \frac{\partial w}{\partial x_j} \right) \frac{\partial v}{\partial x_i} dx \\ b_{\Omega} &= \int_{\Omega} f v dx, \quad b_{\Gamma}(v) = \int_{\Gamma_N} g_N v dS \\ b(v) &= b_{\Omega}(v) + b_{\Gamma}(v). \end{aligned} \quad (8)$$

Z předchozího postupu vyplývá, že je-li u klasickým řešením problému (2) - (4), pak

$$a(u, v) = b(v) \quad \forall v \in V. \quad (9)$$

Naopak se dá dokázat, že pokud funkce u je z prostoru $C^2(\bar{\Omega})$, splňuje (9) a pro každé $x \in \Gamma_D$ platí $u(x) - u_D(x) = 0$, pak je tato funkce klasickým řešením problému (2) - (4).

Nyní již můžeme postoupit k slabé formulaci úlohy. Předpokládejme, že platí (7) a

$$\text{existuje funkce } u_D^* \in X \text{ taková, že } u_D^*|_{\Gamma_D} = u_D. \quad (10)$$

Řekneme, že funkce u je slabým řešením problému (2) - (4), jestliže

$$\begin{aligned} \text{a)} \quad & u \in X, \\ \text{b)} \quad & u - u_D^* \in V, \\ \text{c)} \quad & a(u, v) = b(v) \quad \forall v \in V. \end{aligned} \quad (11)$$

Definice 4 *Budiž*

$$((u, v)) := a(u, v) + b(u, v).$$

Řekneme, že funkce $u \in W^{k,2}(\Omega)$ je slabým řešením okrajové úlohy, jestliže

$$i) \quad u - u_0 \in V,$$

$$ii) \quad \forall v \in V : \quad ((u, v)) = \int_{\Omega} f(x)v(x) dx + g(v)$$

Předpokládejme, že data úlohy splňují podmínky (7) a (10). Pomocí Lax-Milgramova lemmatu ukázat, že problém má právě jedno slabé řešení.

Věta 2 (o existenci a jednoznačnosti slabého řešení) Je-li forma $((u, v))$ navíc V -eliptická, tzn. existuje-li $c > 0$ takové, že

$$\forall v \in V : ((v, v)) \geq c \|v\|_{k,2}^2,$$

pak existuje právě jedno slabé řešení okrajové úlohy, které spojitě závisí na datech, tj.

$$\exists C > 0 : \|u\| \leq C (\|f\| + \|g\|).$$

Důkaz viz [6].

2.3 Metoda konečných prvků

Jak už bylo zmíněno, existuje velmi široká třída PDR, které neumíme řešit analyticky. V těchto případech se využívají numerické metody řešení, jejichž princip spočívá v nahrazení analytického řešení úlohy přibližnými hodnotami tohoto řešení ve vybraných bodech definičního oboru. Takové body obvykle nazýváme uzly sítě. Hodnoty v těchto bodech jsou řešením soustavy jiných než parciálních diferenciálních rovnic. Nejčastěji se jedná o soustavu lineárních rovnic, jejichž sestavení podstatným způsobem závisí na volbě uzlů sítě.

Metoda konečných prvků (MKP) je jednou z nejpoužívanějších numerických metod pro řešení parciálních diferenciálních rovnic. Základní myšlenkou MKP při řešení PDR 2. řádu je rozložení oblasti $\Omega \subseteq \mathbb{R}^2$ na menší části - konečné prvky - na kterých je analýza chování poměrně jednoduchá. Nejčastěji jsou elementy trojúhelníkové a takové rozložení pak nazýváme triangulace. Celý algoritmus MKP se dá vyjádřit v několika krocích:

1. Rozdělení oblasti na podoblasti, tzv. konečné prvky či elementy.
2. Formulace chování jednotlivých elementů.
3. Opětovné složení a získání výsledné soustavy algebraických rovnic s neznámými u_1, \dots, u_n , kde u_i je přibližná hodnota řešení v i -tém uzlu sítě a n je počet uzlů.
4. Aplikace okrajových (či počátečních) podmínek.
5. Vlastní řešení diskretizačních rovnic a získání hodnot řešení v uzlových bodech. Za přibližné řešení rovnice na celé oblasti Ω považujeme destičkovou plochu danou těmito hodnotami.

Diskretizace problému:

Při diskretizaci postupujeme tak, že prostory X a V nahradíme jejich vhodnými podprostory X^h a $V^h = \{v^h \in X^h; v^h|_{\Gamma_D} = 0\}$ konečné dimenze. Přibližným řešením problému pak nazýváme funkci $u^h \in X^h$, pro kterou platí

$$u^h - u_D^{h*} \in V^h \quad \text{a} \quad a(u^h, v^h) = b(v^h) \quad \forall v^h \in V^h, \quad (12)$$

kde $u_D^{h*} \in X^h$ je vhodná aproximace funkce u_D^* .

Dále předpokládáme, že je Ω polygonální. Pro jednoduchost budeme navíc uvažovat jen homogenní Dirichletovu okrajovou podmínku. Pak můžeme položit $u_D^* = u_{Dh}^* \equiv 0$ na $\bar{\Omega}$, čímž problém upravíme na úlohu najít funkci u takovou, že

$$u \in V \quad \text{a} \quad a(u, v) = b(v) \quad \forall v \in V, \quad (13)$$

resp. najít funkci u^h takovou, že

$$u^h \in V^h \quad \text{a} \quad a(u^h, v^h) = b(v^h) \quad \forall v^h \in V^h, \quad (14)$$

Položíme $N = N(h) = \dim(X^h)$, $M = M(h) = \dim(V^h)$. Mějme bázi $\{\varphi_{h1}, \dots, \varphi_{hN}\}$ prostoru X^h očíslovanou takovým způsobem, že funkce $\varphi_{h1}, \dots, \varphi_{hM}$ tvoří bázi prostoru V^h . Necht $(\beta_{h1}, \dots, \beta_{hM})$ jsou souřadnice přibližného řešení vzhledem k uspořádané bázi $(\varphi_{h1}, \dots, \varphi_{hM})$, tj.

$$u^h = \sum_{j=1}^M \beta_{hj} \varphi_{hj}. \quad (15)$$

Protože $a(\cdot, \cdot)$ je bilineární forma na V^h a $b(\cdot)$ lineární funkcionál na V^h , je řešení diskrétního problému (14) ekvivalentní s řešením úlohy najít M -tici reálných čísel $(\beta_{h1}, \dots, \beta_{hM})$ takovou, že platí

$$\sum_{j=1}^M \beta_{hj} a(\varphi_{hj}, \varphi_{hi}) = b(\varphi_{hi}) \quad \forall i \in \{1, \dots, M\} \quad (16)$$

Tím jsme náš problém převedli na řešení soustavy M lineárních rovnic o M neznámých.

Vzhledem k tomu, že M je obvykle velmi velké, je potřeba uvedenou soustavu řešit některou z numerických metod. Bude přitom výhodné, když matice soustavy (16)

$$\mathbf{A} = (a(\varphi_{hj}, \varphi_{hi}))_{i,j=1}^M$$

(tzv. matice tuhosti) bude hodně řídká. Toho můžeme dosáhnout vhodnou volbou podprostorů X^h a jejich bází. A právě na vhodné volbě těchto podprostorů a jejich bází je postavena metoda konečných prvků.

Výběr prostorů X_h a jejich bází:

Všimněme si nejdříve, že platí

$$\begin{aligned} a(\varphi_{hk}, \varphi_{hl}) &= \int_{\Omega} \sum_{i,j=1}^2 \rho_{ij} \frac{\partial \varphi_{hk}}{\partial x_j} \frac{\partial \varphi_{hl}}{\partial x_i} dx \\ &= \int_{\text{sub}(\varphi_{hk} \cap \text{supp}(\varphi_{hl}))} \sum_{i,j=1}^2 \rho_{ij} \frac{\partial \varphi_{hk}}{\partial x_j} \frac{\partial \varphi_{hl}}{\partial x_i} dx, \end{aligned} \quad (17)$$

kde pro $\varphi : \bar{\Omega} \rightarrow \mathbb{R}$ značí $\text{supp}(\varphi)$ nosič funkce φ , tj. uzávěr množiny všech $x \in \bar{\Omega}$, pro která $\varphi(x) \neq 0$. Naším cílem je proto mít takové báze funkce $\varphi_{h1}, \dots, \varphi_{hM}$, že většina dvojic $\varphi_{hk}, \varphi_{hl}$ má prázdný průnik nosičů. Pro tyto dvojice pak bude platit

$$a(\varphi_{hk}, \varphi_{hl}) = 0,$$

čili odpovídající prvek matice soustavy \mathbf{A} bude nulový. Prostory V^h a jejich báze můžeme sestavit následujícím způsobem:

Uvažujme triangulace \mathcal{T}^h uzávěru $\bar{\Omega}$ oblasti Ω , které jsou tvořeny uzavřenými trojúhelníky T a mají tyto vlastnosti

1. $\bigcup_{T \in \mathcal{T}^h} T = \bar{\Omega}$
2. jsou-li $T \neq \tilde{T}$ dva trojúhelníky z \mathcal{T} , pak $T \cap \tilde{T}$ je buď prázdná množina nebo společná strana trojúhelníků T a \tilde{T} nebo jejich společný vrchol,
3. označíme-li E^h množinu všech vrcholů trojúhelníků z \mathcal{T}^h pak $\Gamma_D \cap \Gamma_N \subset E^h$, tj. každý bod, ve kterém se stýkají části hranice s předepsanou Dirichletovou a Neumannovou podmínkou, je vrcholem nějakého trojúhelníka z \mathcal{T}^h .

Prostor X^h nyní můžeme definovat např. takto:

$$X^h = \left\{ v^h \in C(\bar{\Omega}); v^h|_T \text{ je lineární na každém } T \in \mathcal{T}^h \right\}. \quad (18)$$

Spojitosť funkcí z X^h nám zaručí, že budou platit inkluze $X^h \subset X = H^1(\Omega)$, $V^h \subset V$. Snadno lze ověřit, že každá funkce z prostoru X^h je jednoznačně určena svými hodnotami ve vrcholech triangulace (tj. hodnotami v bodech z množiny E^h). Očíslujeme-li tedy vrcholy triangulace takovým způsobem, že vrcholy P_1, \dots, P_M neleží na Γ_D a vrcholy P_{M+1}, \dots, P_N na Γ_D leží, pak můžeme za bázi v prostoru X^h zvolit funkce $\varphi_{h1}, \dots, \varphi_{hN}$ takové, že

$$\varphi_{hi}(P_j) = \delta_{ij} \quad \text{pro } i, j = 1, \dots, N. \quad (19)$$

Funkce $\varphi_{h1}, \dots, \varphi_{hM}$ přitom budou tvořit bázi prostoru V^h .

Vnitřky nosičů báзовých funkcí φ_{hk} a φ_{hl} mají neprázdný průnik jen když P_k a P_l jsou vrcholy jednoho trojúhelníka. Tedy jen v tomto případě je

$$a(\varphi_{hk}, \varphi_{hl}) \neq 0$$

a prvek na pozici (k, l) v matici soustavy \mathbf{A} je nenulový. Při velkém počtu trojúhelníků v triangulaci \mathcal{T}^h tak bude matice \mathbf{A} řídká. Budou-li navíc vrcholy triangulace vhodně očíslovány, bude matice \mathbf{A} pásová. To umožní volit efektivní metody řešení soustavy (16).

Vlastnosti matice tuhosti \mathbf{A} :

Matice tuhosti, se kterou budeme nadále pracovat, má následující tři vlastnosti:

1. Matice je symetrická, tj. $A_{kl} = A_{lk}$.
2. Matice je pásová, tzn. nenulové členy se vyskytují pouze v páse okolo hlavní diagonály.
3. Matice je pozitivně definitní, tj. pro každý vektor x platí $x^T A x \geq 0$, přičemž hodnotu nula tento výraz nabyde pouze v případě, že $x = 0$.

První dvě vlastnosti vedou k faktu, že nemusíme pracovat s celou maticí, ale pouze s diagonálními a naddiagonálními prvky ležícími v páse nad hlavní diagonálou.

Dá se ukázat, že za výše popsanych podmínek je horní odhad čísla podmíněnosti matice tuhosti \mathbf{A} roven

$$\kappa(\mathbf{A}) = \mathbf{O}(\mathbf{h}^{-2}).$$

Poznámka:

Jak bylo zmíněno, při aplikaci MKP nejčastěji diskretizujeme oblast Ω pomocí trojúhelníkových podoblastí. Tato volba má své opodstatnění, pomocí trojúhelníkových podoblastí jsme totiž schopni poměrně dobře popsat i složitější oblast. Z tohoto důvodu, je MKP v textu výše popsána za předpokladu triangulace. Nebudeme se hlouběji zabývat nutnými vlastnostmi takových podoblastí, protože při numerických testech, které budou popsány později, vždy volíme diskretizaci ve formě čtverců. Nicméně, pro všechny výše uvedené postupy a důsledky, postupujeme v případě čtvercové diskretizace analogicky.

3 Metoda rozložení oblasti ve 2D

3.1 Teoretický základ

Následující text čerpá z [1].

Metoda rozložení oblasti (DDM - z anglického domain decomposition methods) je jedna z metod, která nám umožňuje optimalizovat výpočetní náročnost hledání přibližného řešení u . V dnešní době má tato metoda široké využití především díky své snadné paralelizovatelnosti.

Cílem metody rozložení oblasti je najít paralelní SPD předpodmínění $\hat{\mathbf{A}}^{-1}$ tak, aby číslo podmíněnosti $\kappa(\hat{\mathbf{A}}^{-1}\mathbf{A}) = \mathbf{O}\left(\left(1 + \ln \frac{H}{h}\right)^2\right)$.

Metoda rozložení oblasti se zakládá na metodě konečných prvků. Tentokrát však nejprve tvoříme tzv. hrubou síť, jejíž podoblasti následně rozložíme sítí jemnější. Pro názornou ukázkou základní myšlenky DDM, řešme obdobnou úlohu jako v předchozí kapitole:

Uvažujme Poissonovu rovnici s homogenními Dirichletovými podmínkami

$$\begin{aligned} -\operatorname{div}(\rho(x)\nabla u(x)) &= f(x), & x \in \Omega, \\ u(x) &= 0, & x \in \partial\Omega, \end{aligned}$$

kde $\Omega \subset \mathbb{R}^2$ je omezená polygonální oblast s Lipschitzcovskou hranicí, $f \in L^2(\Omega)$ a $\rho \in L^\infty(\Omega)$ jsou pozitivní po částech konstantní materiálové funkce.

Oblast Ω je rozložena na N nepřekrývajících se čtvercových podoblastí Ω_i ve smyslu konečných prvků $\bar{\Omega} = \bigcup_{i=1}^N \bar{\Omega}_i$. Takové rozložení nazýváme hrubá diskretizace či rozložení oblasti. Diskretizace je provedena tak, aby odpovídala změnám materiálové funkce, proto $\rho(x) = \rho_i > 0$ pro $x \in \Omega_i$. Hranice všech podoblastí definujeme jako skeleton $\Gamma := \bigcup_{i=1}^N \bar{E}_i$, kde E_i jsou vnitřní hrany, respektive hranice všech podoblastí kromě $\partial\Omega$, viz Obrázek č. 1. Hrubý diskretizační parametr označujeme $H := \max_{i=1,\dots,N} \operatorname{diam}(\Omega_i)$.

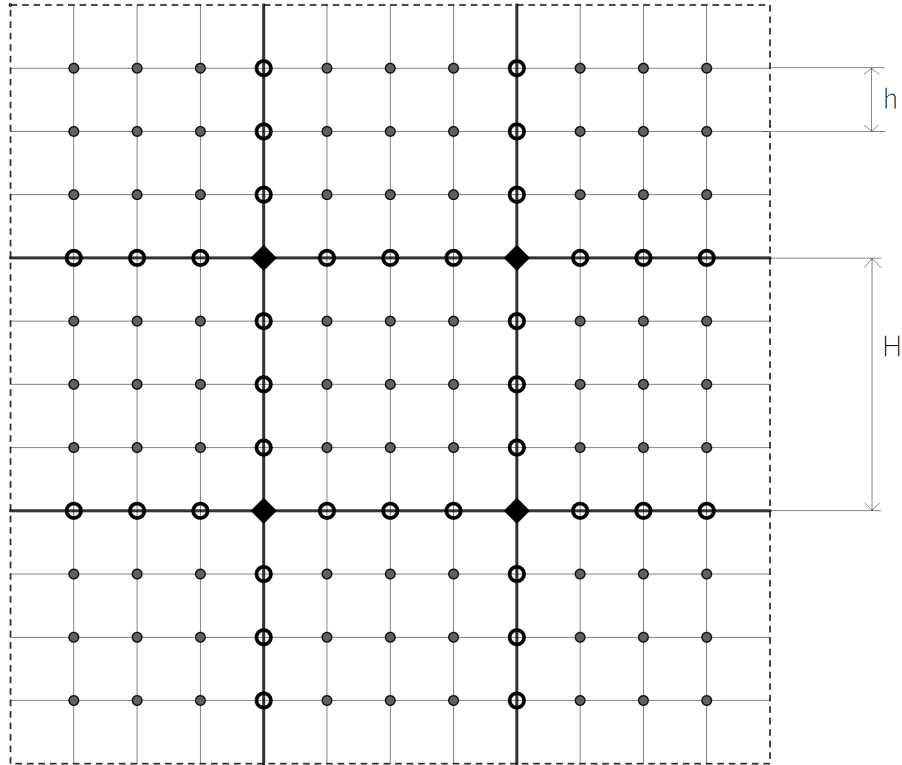
Slabá formulace

$$\text{najdi } u \in H_0^1(\Omega) : \underbrace{\sum_{i=1}^N \rho_i \int_{\Omega_i} \nabla u(x) \nabla v(x) dx}_{=: a(u, v)} = \underbrace{\int_{\Omega} f(x) v(x) dx}_{=: b(v)}, \quad \forall v \in H_0^1(\Omega)$$

je diskretizována příslušnou metodou konečných prvků na podprostoru $V := V^h := \langle \varphi_1(x), \dots, \varphi_n(x) \rangle \subset H_0^1(\Omega)$, kde $(\varphi_i)_{i=1}^n$ jsou Lagrangeovy báze funkce nad uzly, které můžeme vidět na Obrázku č. 1. Jemnou diskretizaci, vycházející z této metody rozložení oblasti, jsme pak schopni popsat a řešit jako soustavu lineárních rovnic

$$\mathbf{A}u = \mathbf{b}, \tag{20}$$

kde $(\mathbf{A})_{i,j} := a(\varphi_i, \varphi_j)$, $(\mathbf{b})_i := b(\varphi_i)$ a $u^h(x) := \sum_{j=1}^n (\mathbf{u})_j \varphi_j(x)$ aproximuje $u(x)$. Diskretizační parametr jemného rozložení oblasti označujeme h a volíme jej jako největší vzdálenost prvků v jemném dělení.



Obrázek 1: Rozložení oblasti Ω na $N \times N = 9$ podoblastí s $n^V = 4$ hraničními uzly hrubé sítě x_i^V (vyznačeno černými kosočtverci); přerušovaná čára označuje $\partial\Omega$; silné plné čáry označují skeleton Γ rozložený na $M = 12$ vnitřních hran E_i s hraničními uzly $x_{i,j}^E$ (vyznačeno prázdnými kruhy); slabé plné čáry označují jemné dělení s $n = 81$ vnitřními uzly $x_{i,j}^I$ (vyznačeno menšími plnými kruhy).

Metoda rozložení oblasti se zakládá na přeuspořádání báзовých funkcí $(\varphi_k^\Gamma)_{k=1}^{n_\Gamma}$ do N systémů funkcí $(\varphi_{i,j}^I)_{j=1}^{n_I^I}$, $i = 1, \dots, N$ nad vnitřními uzly podoblastí jemného dělení $x_{i,j}^I \in \Omega_i$ a jeden systém funkcí $(\varphi_k^\Gamma)_{k=1}^{n_\Gamma}$ nad uzly skeletonu $x_k^\Gamma \in \Gamma \setminus \partial\Omega$, které náleží do množiny uzlů ležících na vnitřních hranách E_i , $x_k^\Gamma = x_{i,j}^E$ nebo do množiny uzlů ležících na hranicích podoblastí hrubého

dělení $x_k^\Gamma = x_i^V$. Touto formou upravíme (20) na sedlobodovou soustavu lineárních rovnic

$$\begin{pmatrix} A_1^{I,I} & 0 & \cdots & 0 & A_1^{I,\Gamma} \\ 0 & A_2^{I,I} & \cdots & 0 & A_2^{I,\Gamma} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & A_N^{I,I} & A_N^{I,\Gamma} \\ A_1^{\Gamma,I} & A_2^{\Gamma,I} & \cdots & A_N^{\Gamma,I} & A^{\Gamma,\Gamma} \end{pmatrix} \begin{pmatrix} u_1^I \\ u_2^I \\ \vdots \\ u_N^I \\ u^\Gamma \end{pmatrix} = \begin{pmatrix} b_1^I \\ b_2^I \\ \vdots \\ b_N^I \\ b^\Gamma \end{pmatrix}, \quad (21)$$

kde $(\mathbf{A}_k^{I,I})_{i,j} := a(\varphi_{k,i}^I, \varphi_{k,j}^I)$, $(\mathbf{A}_k^{I,\Gamma})_{i,j} = (\mathbf{A}_k^{\Gamma,I})_{j,i} := a(\varphi_{k,i}^I, \varphi_{k,j}^\Gamma)$, $(\mathbf{b}_k^I)_i := b(\varphi_{k,i}^I)$ a $(\mathbf{b}^\Gamma)_i := b(\varphi_i^\Gamma)$.

Soustava (21), může být řešena v následujících třech krocích:

1. Vyřešíme MKP soustavu N nezávislých rovnic $\mathbf{A}_i^{\mathbf{I},\mathbf{I}} \mathbf{v}_i^{\mathbf{I}} = \mathbf{b}_i^{\mathbf{I}}$

$$\begin{aligned} -\rho \Delta v_i^I(x) &= f(x), & x \in \Omega_i, \\ v_i^I &= 0, & x \in \partial\Omega_i \end{aligned}$$

na podprostoru $V_i := V_i^h := \langle \varphi_{i,1}^I, \dots, \varphi_{i,n_i}^I \rangle$.

2. Vyřešíme $\mathbf{S} \mathbf{u}^\Gamma = \mathbf{b}^\Gamma - \sum_{i=1}^N \mathbf{A}_i^{\Gamma,\mathbf{I}} \mathbf{v}_i^{\mathbf{I}}$, kde

$$S := A^{\Gamma,\Gamma} - \sum_{i=1}^N A_i^{\Gamma,I} (A_i^{I,I})^{-1} A_i^{I,\Gamma}. \quad (22)$$

3. Vyřešíme MKP soustavu N závislých rovnic $\mathbf{A}_i^{\mathbf{I},\mathbf{I}} \mathbf{w}_i^{\mathbf{I}} = -\mathbf{A}_i^{\mathbf{I},\Gamma} \mathbf{u}^\Gamma$

$$\begin{aligned} -\rho \Delta w_i^I(x) &= 0, & x \in \Omega_i, \\ w_i^I(x) &= u^\Gamma(x), & x \in \partial\Omega_i \cap \Gamma, \\ w_i^I(x) &= 0, & x \in \partial\Omega_i \cap \partial\Omega, \end{aligned}$$

Na tento postup může být nahlíženo také jako na LDL^T faktorizaci

$$A = \begin{pmatrix} I^I & 0 \\ A^{\Gamma,I} (A^{I,I})^{-1} & I^\Gamma \end{pmatrix} \begin{pmatrix} A^{I,I} & 0 \\ 0 & S \end{pmatrix} \begin{pmatrix} I^I & (A^{I,I})^{-1} A^{I,\Gamma} \\ 0 & I^\Gamma \end{pmatrix},$$

kde $\mathbf{I}^{\mathbf{I}}, \mathbf{I}^\Gamma$ jsou jednotkové matice, $\mathbf{A}^{\mathbf{I},\mathbf{I}}$ a $\mathbf{A}^{\mathbf{I},\Gamma} = (\mathbf{A}^{\Gamma,\mathbf{I}})^T$ jsou horní trojúhelníková a dolní trojúhelníková část matice \mathbf{A} v tomto pořadí.

Myšlenkou DD-předpodmínění je nahrazení Schurova doplňku \mathbf{S} jeho aproximací $\hat{\mathbf{S}}$, pro kterou je vytvoření inverzní matice méně náročné, dále číslo podmíněnosti $\kappa(\hat{\mathbf{S}}^{-1} \mathbf{S})$ poroste pomalu pro H/h a je nezávislé na $(\rho_i)_{i=1}^N$.

Předpokládání

V předchozí části textu jsme přeuspořádali báze funkce $(\varphi_i)_{i=1}^n$ do N systémů funkcí vnitřních uzlů a jednoho systému funkcí hraničních uzlů. Podobně teď můžeme přeuspořádat systém báze funkcí $(\varphi_i^\Gamma)_{i=1}^{n^\Gamma}$ nad uzly vnitřních hran $x_{i,j}^E \in E_i$ do M systémů $(\varphi_{i,j}^E)_{j=1}^{n_i^E}$, $j=1, \dots, M$, kde M je počet hran, viz Obrázek č. 1 a do systému funkcí $(\varphi_i^V)_{i=1}^{n^V}$ nad hraničními uzly hrubé sítě $x_i^V \in \Gamma$. Takové přeuspořádání způsobí změny v Schurově doplňku \mathbf{S} ,

$$S = \begin{pmatrix} S^{E,E} & S^{E,V} \\ S^{V,E} & S^{V,V} \end{pmatrix}, \quad (23)$$

kde E-bloky řádků nebo sloupců jsou závislé na funkcích uzlů vnitřních hran $\varphi_{i,j}^E$ a V-bloky jsou závislé na funkcích uzlů hranic hrubé sítě φ_i^V . Matice $\mathbf{S}^{\mathbf{E},\mathbf{E}}$ má následující strukturu

$$S^{E,E} = \begin{pmatrix} S_{1,1}^{E,E} & \cdots & S_{1,M}^{E,E} \\ \vdots & \ddots & \vdots \\ S_{M,1}^{E,E} & \cdots & S_{M,M}^{E,E} \end{pmatrix}, \quad (24)$$

kde $(S_{i,j}^{E,E})_{k,l}$ závisí na změnách v báze funkcích $\varphi_{i,k}^E$ a $\varphi_{j,l}^E$.

Z (22) vidíme, že bloková struktura matice je řídká, protože $\mathbf{S}_{i,j}^{\mathbf{E},\mathbf{E}}$ je nulová, pokud podoblast nesousedí s oběma hranami E_i i E_j .

Označme celkový počet vnitřních hran jako $n^E := \sum_{i=1}^M n_i^E$ a zavedme matici

$$\mathbf{R}^{\mathbf{E}} = (\mathbf{R}_1^{\mathbf{E}}, \dots, \mathbf{R}_M^{\mathbf{E}}) \in \mathbb{R}^{n^V \times n^E}, \quad \mathbf{R}_i^{\mathbf{E}} \in \mathbb{R}^{n^V \times n_i^E},$$

jejíž transpozice je lineární interpolací funkčních hodnot uzlů hranic hrubé diskretizace x_k^V na uzly vnitřních hranic jemné diskretizace $x_{i,j}^E$. To znamená, že hodnoty $\mathbf{R}_i^{\mathbf{E}}$ jsou dány hodnotami báze funkcí hrubé diskretizace

$$(\mathbf{R}_i^{\mathbf{E}})_{\mathbf{k},j} = \varphi_k^H(x_{i,j}^E), \quad (25)$$

kde $(\varphi_i^H)_{i=1}^{n^V}$ jsou funkce metody konečných prvků jednotlivě definované hodnotami uzlů na hranách hrubé diskretizace x_i^V . Vyměnili jsme báze funkce $(\varphi_i^V)_{i=1}^{n^V}$ za $(\varphi_i^H)_{i=1}^{n^V}$, takže Schurův doplněk má tvar

$$S = \begin{pmatrix} I^E & 0 \\ -R^E & I^V \end{pmatrix} \begin{pmatrix} S^{E,E} & \tilde{S}^{E,V} \\ \tilde{S}^{V,E} & \tilde{S}^{V,V} \end{pmatrix} \begin{pmatrix} I^E & -(R^E)^T \\ 0 & I^V \end{pmatrix}, \quad (26)$$

kde $\mathbf{I}^{\mathbf{E}}, \mathbf{I}^{\mathbf{V}}$ jsou jednotkové matice. Nyní blok $\mathbf{A}^{\mathbf{H}} := \tilde{S}^{V,V}$ odpovídá bilineární formě $a(u, v)$ při řešení hrubé diskretizace metodou konečných prvků.

Základní DD-předpodmiňovač je konstruován zanedbáním $\tilde{\mathbf{S}}^{\mathbf{E},\mathbf{V}}, \tilde{\mathbf{S}}^{\mathbf{V},\mathbf{E}}$ a vynecháním mimodiagonálních bloků v (24),

$$\hat{S} = \begin{pmatrix} I^E & 0 \\ -R^E & I^V \end{pmatrix} \begin{pmatrix} \bar{S}^{E,E} & 0 \\ 0 & A^H \end{pmatrix} \begin{pmatrix} I^E & -(R^E)^T \\ 0 & I^V \end{pmatrix},$$

kde $\bar{\mathbf{S}}^{\mathbf{E},\mathbf{E}} := \text{diag}(\mathbf{S}_{1,1}^{\mathbf{E},\mathbf{E}}, \dots, \mathbf{S}_{M,M}^{\mathbf{E},\mathbf{E}})$.

V každé iteraci je pro předpodmíněnou metodu sdružených gradientů zapotřebí určit inverzní matici modifikovaného Schurova doplňku $\hat{\mathbf{S}}^{-1}$, pro kterou známe vzorec

$$\begin{aligned} \hat{S}^{-1} &= \begin{pmatrix} I^E & (R^E)^T \\ 0 & I^V \end{pmatrix} \begin{pmatrix} (\bar{S}^{E,E})^{-1} & 0 \\ 0 & (A^H)^{-1} \end{pmatrix} \begin{pmatrix} I^E & 0 \\ R^E & I^V \end{pmatrix} \\ &= \sum_{i=1}^M \begin{pmatrix} I_i^E \\ 0 \end{pmatrix} (S_{i,i}^{E,E})^{-1} (I_i^E, 0) + \begin{pmatrix} (R_i^E)^T \\ I^V \end{pmatrix} (A^H)^{-1} \underbrace{(R^E, I^V)}_{=: R^H}. \end{aligned} \quad (27)$$

Tento výsledek je modifikací druhého kroku z dříve uvedené „tříkrokové“ metody.

2a. Definujme

$$c^\Gamma := \begin{pmatrix} c^E \\ c^V \end{pmatrix} := b^\Gamma - A^{\Gamma,I} v^I.$$

2b. Řešme M nezávislých lokálních soustav $S_{i,i}^{E,E} w_i^E = c_i^E$.

2c. Řešme globální soustavu hrubého problému $A^H w^H = c^V + R^E c^E$.

2d. Určeme

$$\hat{u}^\Gamma := \begin{pmatrix} w^E + (R^E)^T w^H \\ w^H \end{pmatrix}.$$

Výpočet $\hat{\mathbf{S}}^{-1}$ poskytuje řešení pro globální soustavu matice hrubé diskretizace \mathbf{A}^H a řešení pro M lokálních hraničních úloh s maticemi $\mathbf{S}_{i,i}^{\mathbf{E},\mathbf{E}}$, které jsou lokálními Schurovými doplňky a jsou popsány následujícím systémem rovnic

$$\begin{pmatrix} A_j^{I,I} & 0 & A_{j,i}^{I,E} \\ 0 & A_k^{I,I} & E_{k,i}^{I,E} \\ A_{i,j}^E & A_{i,k}^{E,I} & E_i^{E,E} \end{pmatrix} \begin{pmatrix} w_j^I \\ w_k^I \\ w_i^E \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ c_i^E \end{pmatrix},$$

kde vztahy mezi i, j a k znamenají propojení podoblastí Ω_j a Ω_k hranou E_i . Soustavu rovnic řešíme pro \mathbf{w}_i^E . Jedná se o diskretizaci metodou konečných prvků na podprostorech $V_j + V_k + V_i^E$,

kde $V_i^E := \langle \varphi_{i,l}^E \rangle_{l=1}^{n_i^E}$, následující úlohy na oblasti $\Omega_j \cup \Omega_k$:

$$\begin{aligned} -\rho_j \Delta w_j^I(x) &= 0, & x \in \Omega_j, \\ w_j^I(x) &= 0, & x \in \partial\Omega_j \setminus E_i, \\ -\rho_k \Delta w_k^I(x) &= 0, & x \in \Omega_k, \\ w_k^I(x) &= 0, & x \in \partial\Omega_k \setminus E_i, \\ w_i^E(x) := w_j^I(x) &= w_k^I(x) & x \in E_i, \\ \rho_j \frac{dw_j^I}{dn_j}(x) + \rho_k \frac{dw_k^I}{dn_k}(x) &= c_i^E(x), & x \in E_i, \end{aligned}$$

kde n_j a n_k označují vnější normálové vektory podoblastí Ω_j a Ω_k v tomto pořadí.

Výsledné předpokmínění poskytuje následující faktorizaci

$$\hat{A} = \begin{pmatrix} I^I & 0 \\ A^{\Gamma,I} & (A^{I,I})^{-1} \end{pmatrix} \begin{pmatrix} A^{I,I} & 0 \\ 0 & \hat{S} \end{pmatrix} \begin{pmatrix} I^I & (A^{I,I})^{-1} A^{I,\Gamma} \\ 0 & I^\Gamma \end{pmatrix}. \quad (28)$$

3.2 Vlastní implementace

Nyní popíšeme způsob implementace metody rozložení oblasti:

Pracujeme na čtvercové oblasti $\Omega = \langle 0, 1 \rangle \times \langle 0, 1 \rangle$, rozložené hrubou a jemnou čtvercovou sítí.

Vstupní parametry:

První vstup funkce pro DDM je proměnná N , která udává počet podoblastí hrubé diskretizace $N \times N$. Druhým vstupem je parametr $\frac{H}{h}$, kde $H = \frac{1}{N}$ je diskretizační parametr hrubého dělení a h je diskretizační parametr jemného dělení tak, jak je naznačeno na Obrázku 1. Parametr $\frac{H}{h}$ nám tedy udává, kolik podoblastí jemné diskretizace se bude nacházet v jediné podoblasti hrubé diskretizace. Celkový počet podoblastí jemné diskretizace tak bude $N^2 \times \frac{H}{h}$.

Ilustraci výsledné sítě pro $N = 2$ a $\frac{H}{h} = 3$ můžeme vidět na Obrázku 2.

Lokální matice tuhosti \mathbf{A}_{LOC} :

Při inicializaci uzlů nejprve tvoříme lokální matici jedné podoblasti hrubého dělení. Uzly jedné podoblasti očíslovujeme a rozdělíme do množin vnitřních (na Obrázcích 1,3 a 5 vyznačeno plnými kruhy) a hraničních (na obrázcích vyznačeno prázdnými kruhy, případně kosočtverci) uzlů. Vytvoříme tak lokální matici \mathbf{A}_{LOC} ve smyslu MKP.

Očíslování uzlů jedné podoblasti diskretizace z Obrázku 2 můžeme vidět na Obrázku 3. Jedna podoblast hrubé diskretizace obsahuje celkem 16 uzlů, 4 z nich jsou vnitřní a ostatní leží

na hranicích podoblasti. Výsledná lokální matice \mathbf{A}_{LOC} má rozměry 16×16 a její strukturu, kterou jsme získali pomocí Matlab funkce `spy()`, můžeme vidět na Obrázku 4. Jedná se o blokově diagonální matici, kde čtyři bloky hlavní diagonály popisují čtyři „sloupce“ uzlů z Obrázku 3. Na hlavní diagonále matice \mathbf{A}_{LOC} se nachází hodnoty uzlů 1 až 16. V jednotlivých sloupcích matice jsou pak uchovány hodnoty funkcí pro uzly sousedící s uzlem popsáním hlavní diagonálou. Pokud budeme chtít zkoumat například uzel číslo 5, budeme sledovat pátý sloupec matice. Na hlavní diagonále se nachází funkční hodnota uzlu 5. Ve vybraném sloupci matice pak vidíme, že pozorovaný uzel sousedí s uzly 1, 2, 6, 9 a 10, a to na základě hodnot ležících na řádcích 1, 2, 6, 9 a 10. Díky blokové struktuře máme navíc k dispozici i informaci, že uzly 1, 2 leží na Obrázku 3 ve „sloupci“ před „sloupcem“ obsahujícím uzel pátý, protože se hodnoty funkcí pro tyto uzly nachází v bloku nad blokem obsahujícím pozorovaný uzel. Uzel číslo 6 leží ve stejném „sloupci“, protože se hodnota funkce vyskytuje ve stejném bloku. A uzly 9, 10 se nachází ve „sloupci“ za pátým uzlem, jelikož se funkční hodnoty nachází v bloku pod blokem uzlu číslo 5. Vypovídací hodnota lokální matice je tedy opravdu vyčerpávající.

Globální matice tuhosti \mathbf{A} :

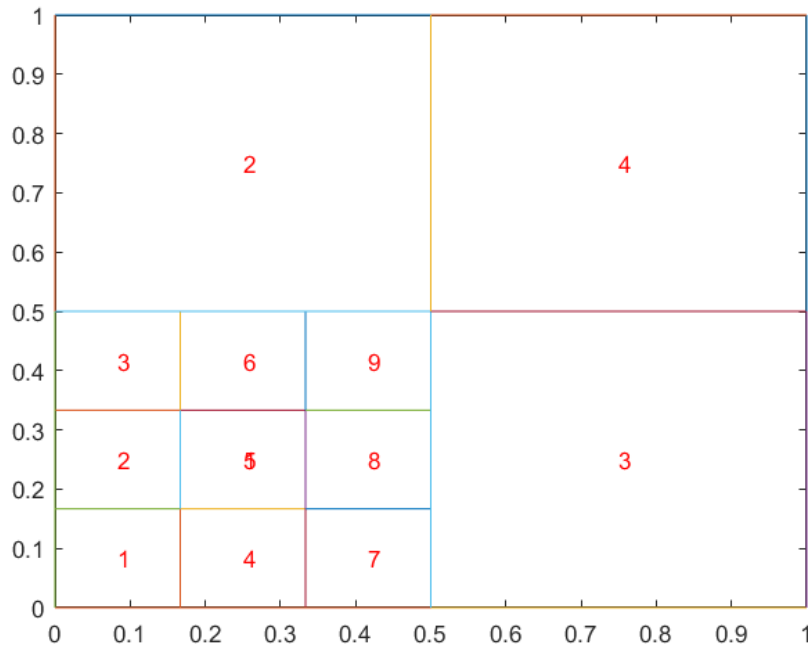
Dále se budeme věnovat globální matici soustavy \mathbf{A} , tedy inicializaci uzlů globální sítě. Zanedbáme hranici $\partial\Omega$ na které je předepsána homogenní Dirichletova okrajová podmínka a zbylé uzly přeuspořádáme do množin vnitřních uzlů ležících na tzv. skeletonu a uzlů hraničních. V případě globální matice je však třeba rozlišit i hraniční uzly podoblastí jemné diskretizace a podoblastí hrubé diskretizace (Obrázcích 1, 3 a 5 vyznačeny kosočtverci). Uzly pak očíslovujeme dle množin do kterých jsme je přeuspořádali (Nejprve vnitřní uzly, poté hraniční uzly jemné diskretizace a nakonec hraniční uzly diskretizace hrubé.). Poté lokální matici \mathbf{A}_{LOC} pomocí interpolační matice „rozšíříme“ dle nových globálních souřadnic uzlů a získáme tak globální matici soustavy \mathbf{A} . Struktura globální matice je popsána v (21).

Očíslování uzlů diskretizace z Obrázku 2 můžeme vidět na Obrázku 5. Diskretizací z Obrázku 2 jsme získali 16 vnitřních uzlů (4 pro každou podoblast hrubé diskretizace), 8 hraničních uzlů jemné diskretizace (2 pro každou hranu hrubé diskretizace) a 1 jeden hraniční uzel hrubé diskretizace, který je v tomto případě společný pro všechny podoblasti hrubé sítě. Celkem tedy síť obsahuje 25 uzlů a globální matice \mathbf{A} má rozměry 25×25 . Strukturu globální matice můžeme vidět na Obrázku 6. Hodnoty uzlů se nám sice trochu „rozutekly“, ale vypovídací hodnota globální matice je stále kvalitní. Hodnoty hlavní diagonály matice \mathbf{A} stejně jako v případě lokální matice popisují hodnoty jednotlivých uzlů a mimodiagonální prvky obsahují funkční hodnotu uzlů sousedních. Čtyři bloky, které jsou z Obrázku 6 vidět na první pohled, popisují vnitřní uzly jednotlivých podoblastí hrubé diskretizace a jejich sousedy, kteří jsou vnitřními uzly stejné podoblasti. Matice je symetrická a hodnoty ve stejném sloupci, ale rozprostřené mimo bloky vnitřních uzlů, popisují jejich sousedy vyskytující se na skeletonu. Hodnota na posledním řádku, či v posledním sloupci říká, že daný uzel sousedí s hraničním uzlem hrubé diskretizace. Uzly hlavní diagonály, které nejsou obsaženy ve čtyřech výrazných blocích obsahují hodnoty uzlů

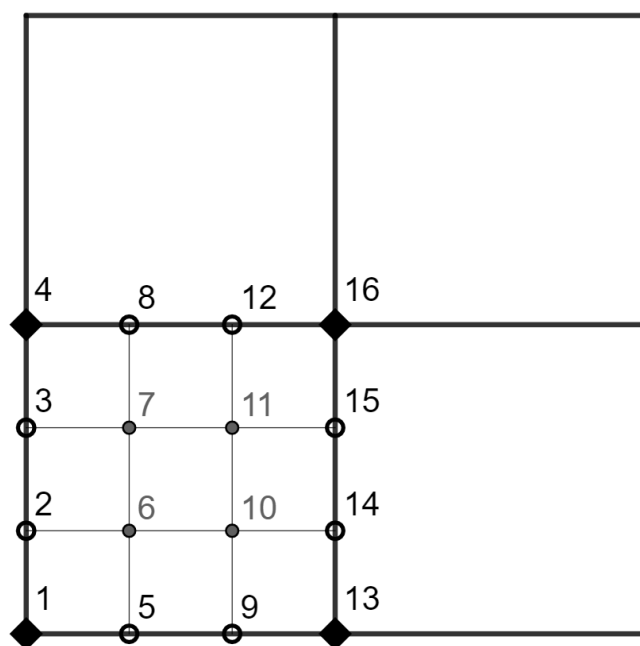
skeletonu a obdobně jako v případě lokální matice, přímo sousedící hodnoty říkají, že se uzel nachází na stejné hranici. Vzdálenější hodnoty pak popisují vztah uzlu skeletonu s uzly vnitřními či hraničním uzlem hrubé diskretizace.

Předpodmínění globální matice tuhosti \mathbf{A}

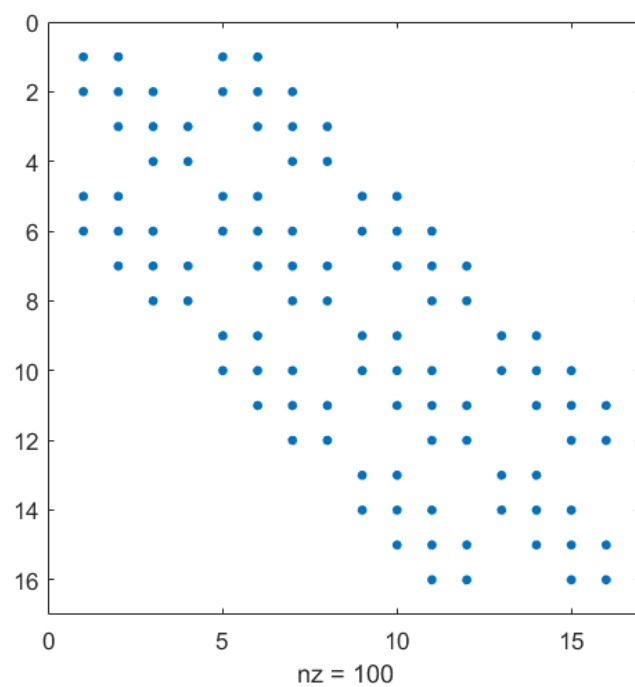
Při výpočtu optimálního předpodmínění globální matice tuhosti \mathbf{A} postupujeme tak, jak je popsáno v „tříkrokové“ metodě z teoretická části této kapitoly.



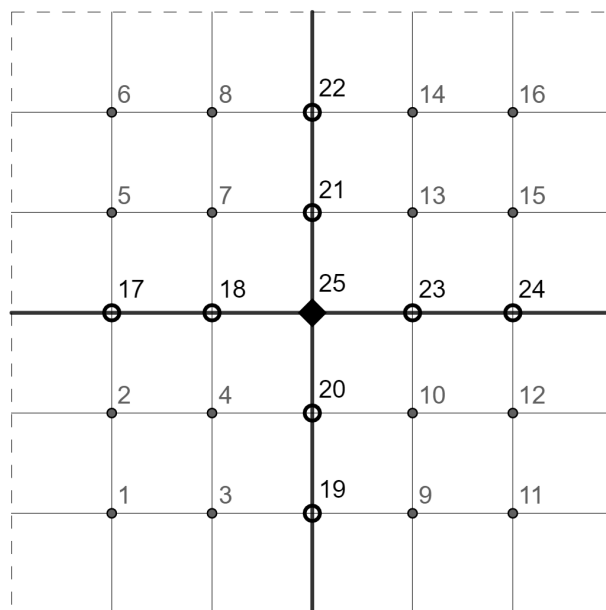
Obrázek 2: Rozložení oblasti $\Omega = \langle 0, 1 \rangle \times \langle 0, 1 \rangle$ pro $N = 2$ a $\frac{H}{h} = 3$.



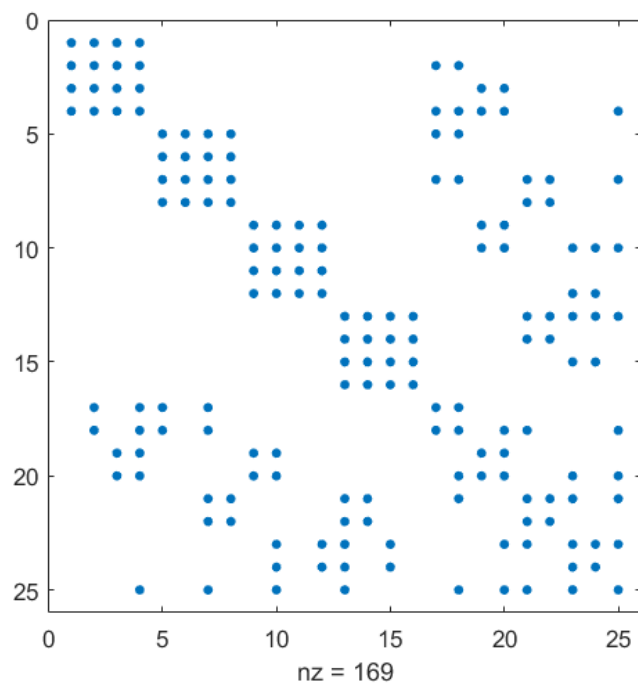
Obrázek 3: Očíslování uzlů jedné podoblasti při tvorbě lokální matice A_{LOC}



Obrázek 4: Struktura lokální matice A_{LOC}



Obrázek 5: Očíslování uzlů oblasti Ω při tvorbě globální matice A



Obrázek 6: Struktura globální matice A

3.3 Numerické experimenty

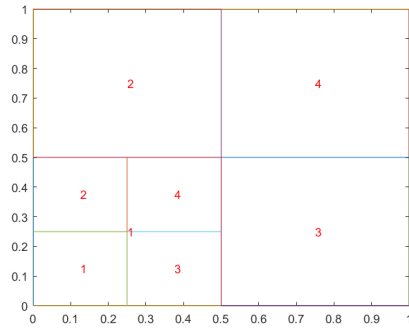
Nyní se budeme věnovat numerickým experimentům na soustavě, kterou jsme vytvořili pomocí DDM. Nejprve se podíváme na vlastnosti čísla podmíněnosti a stabilitu naší soustavy. Poté se pustíme do hledání řešení soustavy, kde budeme sledovat rychlost konvergence pro předpokládanou metodu sdružených gradientů.

3.3.1 Analýza čísla podmíněnosti

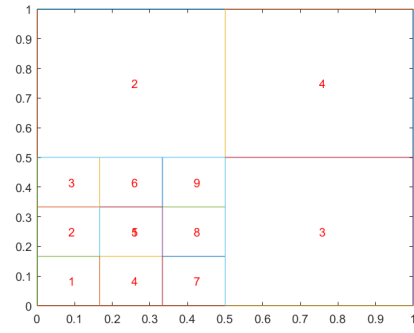
Dá se ukázat, že za výše popsanych podmínek je číslo podmíněnosti $\kappa(\hat{\mathbf{A}}^{-1}\mathbf{A}) \approx \kappa(\hat{\mathbf{S}}^{-1}\mathbf{S})$ zhora ohraničeno hodnotou $c \left(1 + \ln \frac{H}{h}\right)^2$, kde konstanta $c > 0$ a je nezávislá na H a h . Číslo podmíněnosti by tak nemělo být závislé na počtu podoblastí hrubé diskretizace.

Pozorujme chování čísla podmíněnosti $\kappa(\hat{\mathbf{S}}^{-1}\mathbf{S})$ pro různé případy hrubé diskretizace, konkrétně pro $N = 2, 4, 6, \dots, 30$ a konstantní jemné dělení sítě $\frac{H}{h} = 2$ a 3 . Diskretizujeme tedy čtverec o rozměrech $\langle 0, 1 \rangle \times \langle 0, 1 \rangle$ postupně na $N \times N$ podoblastí, a poté každou z těchto podoblastí hrubé diskretizace rozdělíme jemnou sítí na $\frac{H}{h} \times \frac{H}{h}$ podoblastí jemné diskretizace, tak jak je naznačeno na Obrázku 7.

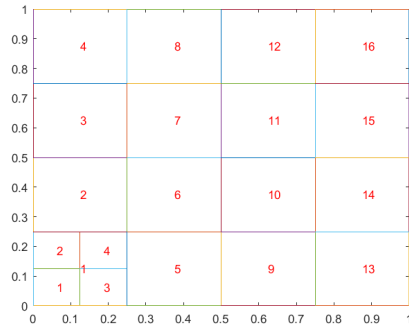
Výsledné hodnoty čísla podmíněnosti $\kappa(\hat{\mathbf{S}}^{-1}\mathbf{S})$ jsou zachyceny na Obrázku 8. V grafu si můžeme všimnout zkracujících se kroků mezi jednotlivými křivkami, a dojít tak k závěru, že se jejich pozice od určitého N nadále nebude měnit, respektive, křivky se ustálí v dostatečně malém okolí jediné linie. Tím jsme podpořili předpoklad, že horní hranice čísla podmíněnosti předpokládané DDM soustavy je nezávislá na hrubé diskretizaci.



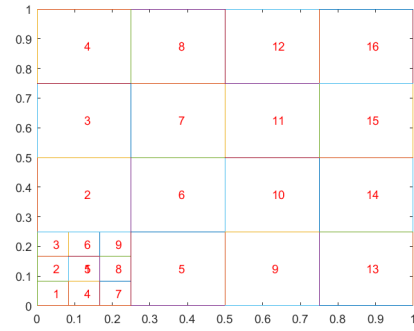
(a) $N = 2, \frac{H}{h} = 2$



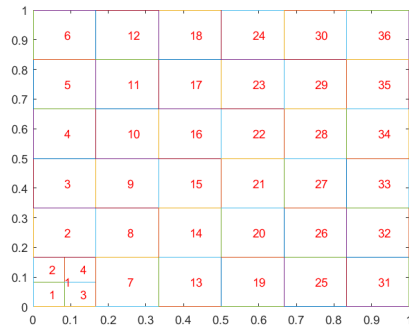
(b) $N = 2, \frac{H}{h} = 3$



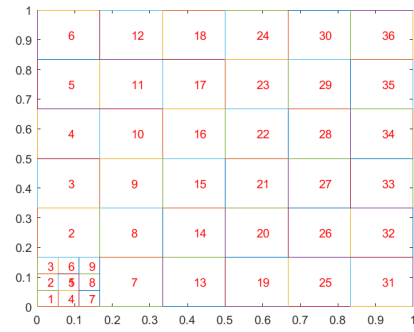
(c) $N = 4, \frac{H}{h} = 2$



(d) $N = 4, \frac{H}{h} = 3$

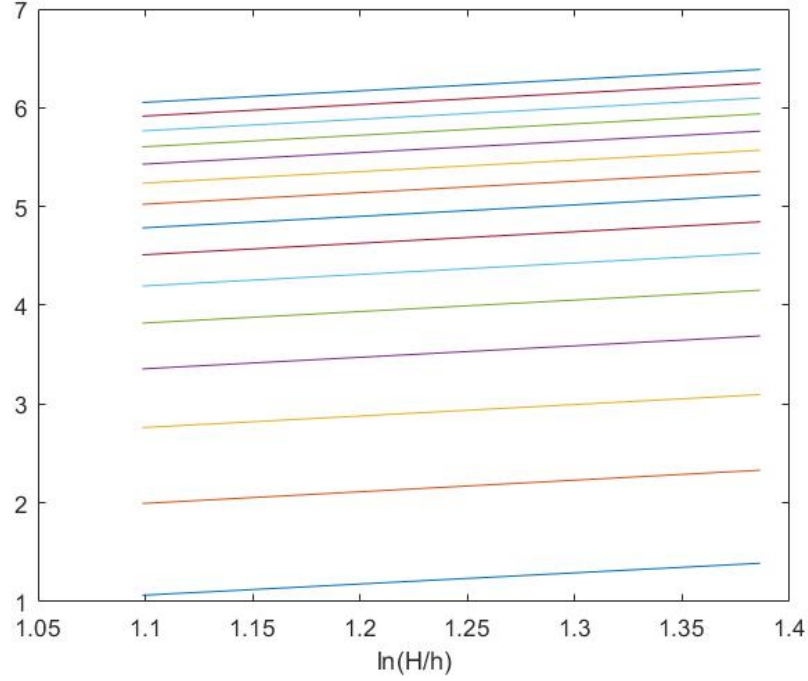


(e) $N = 6, \frac{H}{h} = 2$



(f) $N = 6, \frac{H}{h} = 3$

Obrázek 7: Diskretizace oblasti Ω



Obrázek 8: Číslo podmíněnosti $\kappa(\hat{\mathbf{S}}^{-1}\mathbf{S})$ pro $N = 2, 4, 6, \dots, 30$ a $\frac{H}{h} = 2, 3$ v logaritmickém měřítku

Dále věnujme pozornost stabilitě soustavy. Budeme testovat, jak se změní podmíněnost matice \mathbf{A} při náhodných posunech jejích vlastních čísel, respektive budeme pozorovat změny čísla podmíněnosti $\kappa(\hat{\mathbf{S}}^{-1}\mathbf{S}) \approx \kappa(\hat{\mathbf{A}}^{-1}\mathbf{A})$.

Testování proběhlo posunem vlastních čísel matice \mathbf{A} o 1 až 50 procent náhodně zvolené hodnoty (mezi 0 a 1). Vlastní čísla jsme o náhodnou hodnotu nejprve zvětšovali a poté zmenšovali.

Na Obrázku 9 můžeme vidět, jakým způsobem perturbace změnila číslo podmíněnosti pro $N = 2$ a $\frac{H}{h} = 4$ (matice \mathbf{A} o rozměrech 49×49). V případě těchto vstupních údajů je původní $\kappa(\hat{\mathbf{S}}^{-1}\mathbf{S})$ rovno přibližně 5,14. V obou grafech se maximální odchylka nachází v intervalu $\langle 4, 5; 7 \rangle$, a můžeme tak dojít k závěru, že směr změny vlastních čísel matice \mathbf{A} (přičítání či odčítání náhodné hodnoty) výrazněji neovlivňuje výsledné odchylky. Není překvapením, že se větší výkyvy vyskytují u vyššího koeficientu perturbace. Křivky grafů na Obrázku 9 nejsou zcela stejné, to je samozřejmě způsobeno náhodným výběrem hodnot perturbace. Věnujme se dále jen výsledkům, kdy jsme k vlastním číslům matice náhodné hodnoty přičítali. Nejzajímavějším údajem je pro nás maximální odchylka od původního čísla podmíněnosti, která v tomto případě činí přibližně 1,47.

Proveďme stejné pozorování pro $N = 2$ a $\frac{H}{h} = 16$ (matice \mathbf{A} o rozměrech 961×961), kdy je číslo podmíněnosti neupravené soustavy rovno přibližně 13,86. Na Obrázku 10 můžeme vidět, že se oba grafy opět pohybují ve stejném intervalu a nejvýraznější výkyv v grafu 14a je přibližně

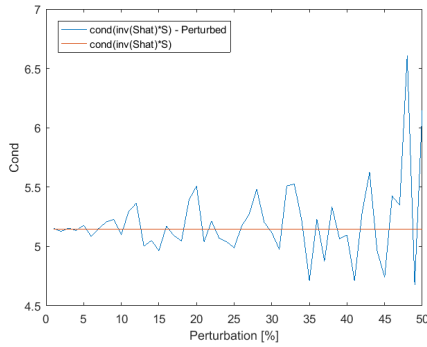
1,36. Získali jsme o něco menší odchylku, než v předchozím případě a výsledky můžeme považovat za příznivé.

Otestujme změnu čísla podmíněnosti i pro $N = 2$ a $\frac{H}{h} = 32$ (matice \mathbf{A} o rozměrech $3\,969 \times 3\,969$), graf hodnot čísla podmíněnosti po přičtení náhodných hodnot k vlastním číslům matice \mathbf{A} můžeme vidět na Obrázku 11. Maximální odchylka zde činí 0,73, opět jsme dosáhli o něco nižšího čísla než v předchozích případech.

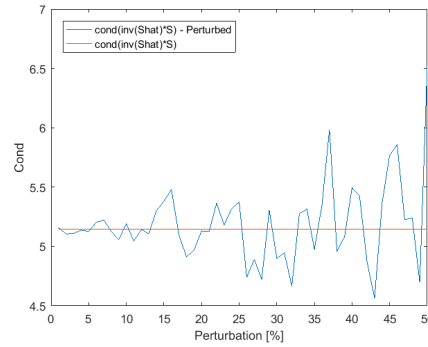
Nyní se zdá, že čím jemnější je síť jemné diskretizace, tím jsou odchylky menší. Abychom ověřili, zda vyšší parametr $\frac{H}{h}$ znamená méně či více stabilní systém, provedme několik testů pro vstupní údaje $N = 2$, $\frac{H}{h} = 4$ a $N = 2$, $\frac{H}{h} = 16$. V Tabulce 1 můžeme vidět, že při všech testech je odchylka pro nižší $\frac{H}{h}$ vždy menší, a dojít tak k závěru, že výsledky prvního testování jsou spíše náhodné. Nicméně můžeme položit předpoklad, že i pro poměrně jemnou diskretizaci je soustava stabilní.

Vstupní údaje	Test									
	1	2	3	4	5	6	7	8	9	10
$N = 2, \frac{H}{h} = 4$	0,56	0,63	0,73	0,80	0,65	0,51	1,30	0,56	0,91	0,94
$N = 2, \frac{H}{h} = 16$	0,73	1,12	1,64	1,71	1,35	1,21	1,04	1,09	1,24	1,41

Tabulka 1: Maximální odchylka čísla podmíněnosti $\kappa(\hat{\mathbf{S}}^{-1}\mathbf{S})$ po perturbaci matice \mathbf{A} pro různé parametry $\frac{H}{h}$

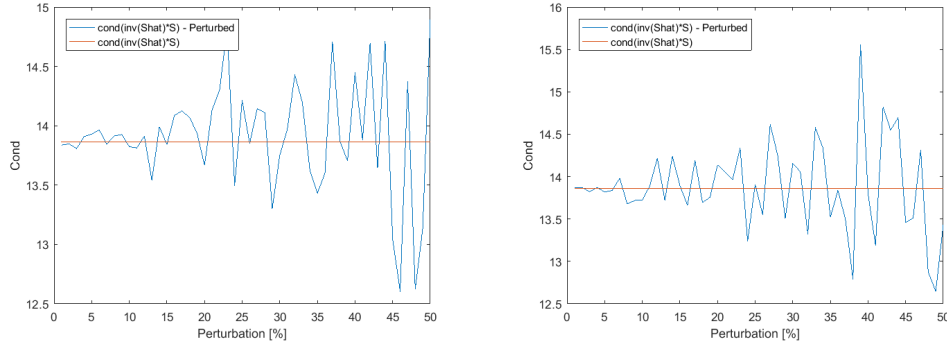


(a) Změna čísla podmíněnosti $\kappa(\hat{\mathbf{S}}^{-1}\mathbf{S})$ po přičtení náhodných hodnot k vlastním číslům matice \mathbf{A} (modrá) a původní číslo podmíněnosti (červená).



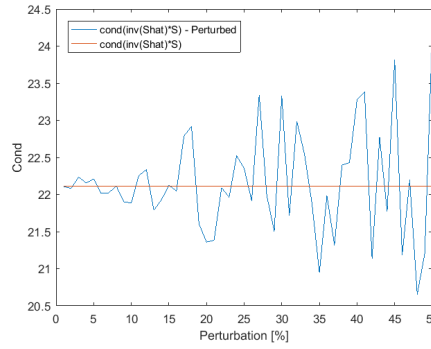
(b) Změna čísla podmíněnosti $\kappa(\hat{\mathbf{S}}^{-1}\mathbf{S})$ po odečtení náhodných hodnot k vlastním číslům matice \mathbf{A} (modrá) a původní číslo podmíněnosti (červená)

Obrázek 9: Perturbace matice \mathbf{A} pro $N = 2$, $\frac{H}{h} = 4$.



(a) Změna čísla podmíněnosti $\kappa(\hat{\mathbf{S}}^{-1}\mathbf{S})$ po přičtení náhodných hodnot k vlastním číslům matice \mathbf{A} (modrá) a původní číslo podmíněnosti (červená) (b) Změna čísla podmíněnosti $\kappa(\hat{\mathbf{S}}^{-1}\mathbf{S})$ po odečtení náhodných hodnot k vlastním číslům matice \mathbf{A} (modrá) a původní číslo podmíněnosti (červená)

Obrázek 10: Perturbace matice \mathbf{A} pro $N = 2$, $\frac{H}{h} = 16$.



Obrázek 11: Perturbace matice \mathbf{A} pro $N = 2$, $\frac{H}{h} = 16$. Změna čísla podmíněnosti $\kappa(\hat{\mathbf{S}}^{-1}\mathbf{S})$ po přičtení náhodných hodnot k vlastním číslům matice \mathbf{A} (modrá) a původní číslo podmíněnosti (červená)

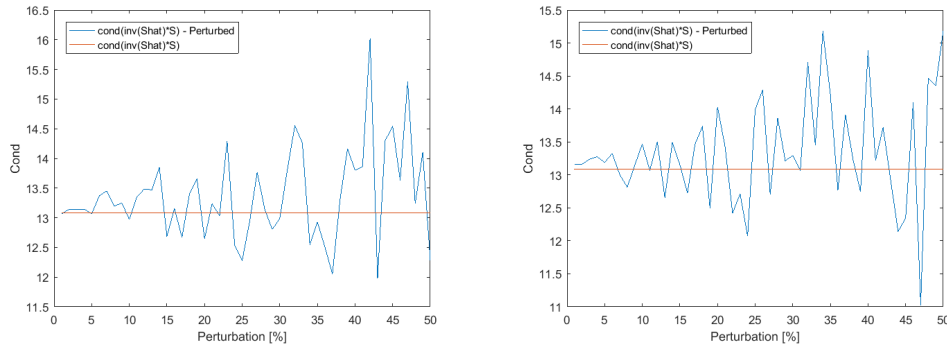
Nezapomeňme na závěr testování stability DDM soustavy ověřit i výsledky pro vyšší N . Podíváme se na výsledky pro $N = 4, 6, 8$ a $\frac{H}{h} = 4$ (matice \mathbf{A} o rozměrech 225×225 , 529×529 a 961×961).

Výsledky můžeme vidět na Obrázcích 12, 13 a 14. Maximální odchylka pro $N = 4$ činí 2,93, pro $N = 6$ je to 7,73 a pro $N = 8$ dokonce 14,17. Výkyvy hodnot čísla podmíněnosti pro vyšší počet podoblastí hrubé diskretizace výrazně stoupají (Tabulka 2 potvrzuje, že se nejedná o náhodný jev.) a intervaly výkyvů pro posun vlastních čísel po přičtení a odečtení náhodných hodnot už si nejsou tolik podobné. Celkově se zdá, že pro tyto úlohy, je soustava méně stabilní. Není však třeba klesat na mysli. Z této kapitoly víme, že číslo podmíněnosti $\kappa(\hat{\mathbf{S}}^{-1}\mathbf{S})$ je nezávislé na počtu podoblastí hrubého dělení, nicméně díky výsledkům z předchozího testování také víme, že se tato skutečnost projeví až pro poměrně vysoká N . Pro vyšší N se číslo podmíněnosti

Vstupní údaje	Test									
	1	2	3	4	5	6	7	8	9	10
$N = 4, \frac{H}{h} = 4$	2,76	2,62	3,25	2,86	2,93	2,08	2,75	3,21	3,72	1,64
$N = 6, \frac{H}{h} = 4$	9,00	8,61	5,59	6,88	7,78	7,99	6,56	7,00	5,01	7,97

Tabulka 2: Maximální odchylka čísla podmíněnosti $\kappa(\hat{\mathbf{S}}^{-1}\mathbf{S})$ po perturbaci matice \mathbf{A} pro měnící se parametr N

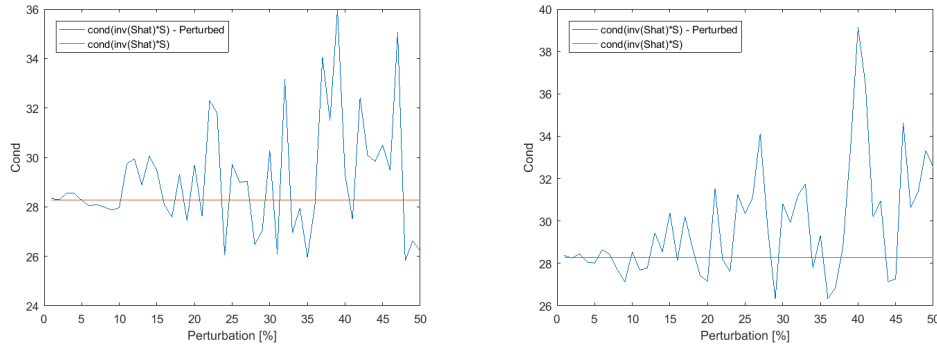
přestane měnit a stabilita soustavy tak bude záviset jen na proměnné $\frac{H}{h}$, jejíž zvyšování, jak už jsme si ukázali, nezpůsobuje při perturbaci velké změny v podmíněnosti soustavy.



(a) Změna čísla podmíněnosti $\kappa(\hat{\mathbf{S}}^{-1}\mathbf{S})$ po přičtení náhodných hodnot k vlastním číslům matice \mathbf{A} (modrá) a původní číslo podmíněnosti (červená)

(b) Změna čísla podmíněnosti $\kappa(\hat{\mathbf{S}}^{-1}\mathbf{S})$ po odečtení náhodných hodnot k vlastním číslům matice \mathbf{A} (modrá) a původní číslo podmíněnosti (červená)

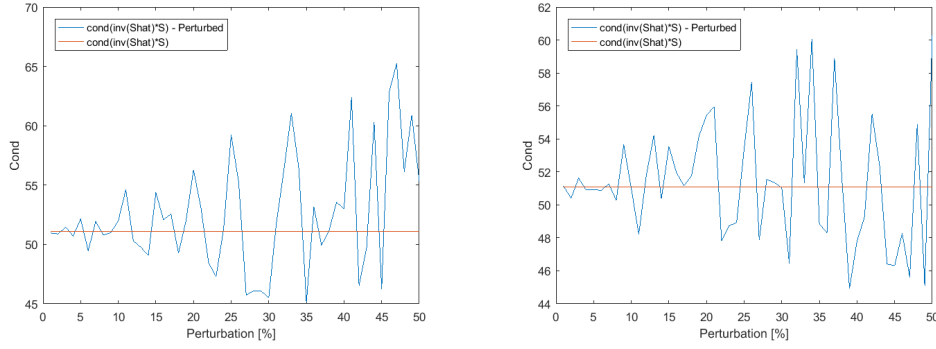
Obrázek 12: Perturbace matice \mathbf{A} pro $N = 4, \frac{H}{h} = 4$.



(a) Změna čísla podmíněnosti $\kappa(\hat{\mathbf{S}}^{-1}\mathbf{S})$ po přičtení náhodných hodnot k vlastním číslům matice \mathbf{A} (modrá) a původní číslo podmíněnosti (červená)

(b) Změna čísla podmíněnosti $\kappa(\hat{\mathbf{S}}^{-1}\mathbf{S})$ po odečtení náhodných hodnot k vlastním číslům matice \mathbf{A} (modrá) a původní číslo podmíněnosti (červená)

Obrázek 13: Perturbace matice \mathbf{A} pro $N = 6, \frac{H}{h} = 4$.



(a) Změna čísla podmíněnosti $\kappa(\hat{\mathbf{S}}^{-1}\mathbf{S})$ po přičtení náhodných hodnot k vlastním číslům matice \mathbf{A} (modrá) a původní číslo podmíněnosti (červená) (b) Změna čísla podmíněnosti $\kappa(\hat{\mathbf{S}}^{-1}\mathbf{S})$ po odečtení náhodných hodnot k vlastním číslům matice \mathbf{A} (modrá) a původní číslo podmíněnosti (červená)

Obrázek 14: Perturbace matice \mathbf{A} pro $N = 8$, $\frac{H}{h} = 4$.

V dalších testech budeme hledat řešení u soustavy lineárních rovnic $Au = b$ pomocí iterační metody zvané předpodmíněná metoda sdružených gradientů, proto se jí nyní krátce věnujme.

3.3.2 Předpodmíněná metoda sdružených gradientů

Následující text čerpá z [2] a [3].

Podstatou metody sdružených gradientů (CG), jejíž algoritmus můžeme vidět níže (Algoritmus 1), je najít nejlepší aproximaci (v A -normě) v daném Krylovově podprostoru.

Ačkoliv posloupnost aproximací u_k konverguje k řešení u rychle, rychlost konvergence můžeme ještě navýšit, do velké míry totiž závisí na čísle podmíněnosti matice \mathbf{A} . Pro MKP soustavy se dá dokonce ukázat, že odhad počtu iterací je přímo-úměrný číslu podmíněnosti matice \mathbf{A} . Tato skutečnost vede k myšlence upravit soustavu rovnic tak, aby řešení zůstalo nezměněné a matice nové soustavy měla menší číslo podmíněnosti. Navýšit rychlost konvergence můžeme tedy předpodmíněním soustavy. Zvolme libovolnou SPD matici \mathbf{C} . Soustavu $Au = b$ pak můžeme psát ve tvaru

$$(C^{-1/2}AC^{-1/2})(C^T u) = C^{-1}b.$$

Označíme-li $\hat{A} = C^{-1/2}AC^{-1/2}$, $\hat{u} = C^T u$ a $\hat{b} = C^{-1}b$, můžeme soustavu rovnic vyjádřit rovností

$$\hat{A}\hat{u} = \hat{b},$$

kde matice \hat{A} je SPD. Na soustavu teď aplikujeme CG a získáme aproximaci řešení této soustavy \hat{u}_k . Abychom získali aproximaci u_k původní soustavy $Au = b$, stačí vektor \hat{u} zleva vynásobit

maticí C^{-T} , tedy

$$u_k = C^{-T} \hat{u}_k.$$

Potřebný předpodmiňovač máme k dispozici z „tříkrokové“ metody z předcházejícího textu této kapitoly a můžeme tak při hledání řešení u aplikovat předpodmíněnou metodu sdružených gradientů.

Algoritmus 1 Metoda sdružených gradientů

Vstup: $n, A, b, MaxIter, toler$

```
 $u = \text{zeros}(n, 1)$   
 $r = b - A * u$   
 $v = r$   
for  $k = 1, 2, \dots, MaxIter$  do  
   $a = (v \cdot r) / (v \cdot A * v)$   
   $u = u + a * v$   
   $r = r - a * A * v$   
   $(b = -v \cdot A * r) / (v \cdot A * v)$   
   $v = r + b * v$   
  if  $(\text{norm}(r) < toler)$  then  
    break  
  end if  
end for  
Výstup:  $u$ 
```

3.3.3 Rychlost konvergence

Hledejme řešení u soustavy lineárních rovnic $Au = b$ pomocí předpodmíněné metody sdružených gradientů. Potřebný předpodmiňovač získáme pomocí „tříkrokové“ metody z teoretické části této kapitoly. Sledujme počet iterací provedených k určení řešení soustavy ve vztahu s různými vstupními parametry N a $\frac{H}{h}$.

V podkapitole o předpodmíněné metodě sdružených gradientů jsme zmínili, že počet iterací této metody je v případě MKP matice přímo-úměrný číslu podmíněnosti matice \mathbf{A} . Vzhledem k tomu, že číslo podmíněnosti $\kappa(\hat{\mathbf{S}}^{-1}\mathbf{S})$ je (jak už jsme si ukázali) od určitého okamžiku nezávislé na počtu podoblastí hrubého dělení dá se v tomto okamžiku očekávat i ustálení počtu iterací potřebných k výpočtu řešení u .

Pokud nyní nahlédneme do Tabulek 3 a 4, ve kterých je zachycen počet iterací předpodmíněné metody sdružených gradientů nejprve pro $N = 2, 4, 6, \dots, 30$ a $\frac{H}{h} = 4$, a poté pro $N = 2, 4, 6, \dots, 30$ a $\frac{H}{h} = 16$, vidíme, že je náš předpoklad potvrzen. V konkrétním okamžiku se počet iterací přestane měnit v závislosti na zvyšujícím se N , a my tak budeme od tohoto okamžiku pro libovolně vysoký počet podoblastí hrubé diskretizace provádět stále stejný počet iterací.

$\frac{H}{h} = 4, N =$	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Počet iterací	5	5	10	9	11	12	12	12	11	11	11	11	11	10	10
$\frac{H}{h} = 4, N =$	17	18	19	20	21	22	23	24	25	26	27	28	29	30	
Počet iterací	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10

Tabulka 3: Počet iterací předpodmíněné metody sdružených gradientů pro $\frac{H}{h} = 4$ a měnící se parametr N

$\frac{H}{h} = 16, N =$	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Počet iterací	6	6	12	11	14	13	13	13	13	13	13	13	13	13	13
$\frac{H}{h} = 16, N =$	17	18	19	20	21	22	23	24	25	26	27	28	29	30	
Počet iterací	13	13	13	13	13	13	13	13	13	13	13	13	13	13	13

Tabulka 4: Počet iterací předpodmíněné metody sdružených gradientů pro $\frac{H}{h} = 16$ a měnící se parametr N

Pomocí DDM od určitého počtu podoblastí hrubé diskretizace vytváříme dobře podmíněnou stabilní soustavu. Díky předpodmiňovače, který nám DDM poskytuje, jsme navíc schopni efektivně hledat řešení soustavy předpodmíněnou metodou sdružených gradientů, kdy rychlost konvergence téměř nezávisí na počtu podoblastí hrubé diskretizace. Nicméně algoritmus konstrukce předpodmiňovače - pomocí (modifikované) „tříkrokové“ metody z teoretické části této kapitoly - je z důvodu využití Gaussovy eliminace - při řešení MKP soustav v prvním a třetím kroku - výpočetně náročný. V následující kapitole si proto popíšeme metodu, která nám později pomůže hledat řešení ještě efektivněji.

4 Multigrid ve 2D

4.1 Teoretický základ

Multigrid je numerická metoda, kterou můžeme využít nejen k hledání řešení diferenciálních rovnic, ale i jako předpodmiňovač. Dokonce se dá ukázat, že v Krylovově podprostoru je efektivnější využít multigrid jako předpodmiňovač, než jako řešič. Neméně důležitá vlastnost multigridu je, že může být kombinován i s jinými diskretizačními metodami (Např. s metodou konečných objemů, metodou sítí a v neposlední řadě s MKP). Při hledání řešení MKP soustav je multigrid nejrychlejší dnes známá metoda.

Základní myšlenkou multigridu je urychlit konvergenci aplikované iterační metody aproximací jemné sítě na síť hrubší, na které je hledání řešení výpočetně méně náročné. Tento rekurzivní algoritmus je opakován, dokud nedosáhneme takové soustavy, jejíž náročnost přímého řešení je téměř zanedbatelná ve srovnání s náročností řešení na původní jemné síti. Řešením hrubého problému, tak rychle dostaneme dobrý počáteční iterační odhad při hledání řešení soustavy.

4.1.1 Algoritmus metody dvou sítí

Při implementaci jsme využili takzvané two-level/two-grid metody, česky zvané metoda dvou sítí. Ta neprobíhá rekurzivně, ale vyžívá jen jediné tvorby hrubší sítě. Multigrid o n -úrovních je samozřejmě efektivnější, nicméně našim úkolem je nahradit Gaussovu eliminaci, využívanou při výpočtech, nějakou „levnější“ metodou, a proto si nyní vystačíme s metodou dvou sítí.

Algoritmus metody dvou sítí je následující:

1. Vyhlazení (Pre - Smoothing)

Několik iterací jednoduché iterační metody (v našem případě Richardsonova metoda) nám poskytne aproximaci řešení u^h na původní jemné síti a residuum r^h na původní jemné síti.

2. Prolongace

Pomocí matice prolongace \mathbf{P} provedeme:

- (a) Interpolaci matice původní jemné sítě A^h na matici nové hrubé sítě A^H .

$$A^H = P^T \cdot A^h \cdot P$$

- (b) Interpolaci residua původní jemné sítě r^h na residuum nové hrubé sítě r^H .

$$r^H = P^T \cdot r^h$$

- (c) V tomto kroku využíváme Gaussovy eliminace aplikované na matici A^H nové hrubé sítě. Jedná se o aproximaci chyby nové hrubé sítě $e^H = A^H \setminus r^H$ na chybu původní

jemné síti e^h .

$$e^h = P \cdot e^H$$

3. Upravíme aproximaci řešení u^h přičtením vypočtené chyby e^h

$$u^h = u^h + e^h$$

a vypočteme nové residuum jako

$$r^h = r^h - A^h \cdot e^h$$

4. Vyhlazení (Post - Smoothing)

Po úpravě aproximace řešení u^h můžeme znova provést několik iterací zvolené iterační metody.

Matice prolongace

Základem výše popsaného algoritmu je sestavení tzv. matice prolongace \mathbf{P} . Ta pro nás vytváří „cestu“ mezi původní jemnou a novou hrubou sítí.

Pro jednoduchost si sestavení takové matice nejprve demonstrujeme na jednodimensionálním příkladě. Na Obrázku 15 můžeme vidět tři uzly a ilustraci jejich bazových funkcí, tedy demonstraci hrubé sítě, které bychom rádi dosáhli. Zaměříme se na uzel číslo dvě, jeho bazovou funkci popíšeme vektorem $(0, 1, 0)^T$, kdy první složka vektoru, odpovídá hodnotě bazové funkce nad uzlem číslo jedna, druhá složka hodnotě bazové funkce v bodě dva a tak dále. Na Obrázku 16 jsme přidali dva uzly, a vytvořili tak původní jemnější síť. Pokud bychom chtěli sestavit sloupec matice prolongace, který odpovídá uzlu číslo 3 na jemné síti a u uzlu číslo 2 na hrubé síti, měl by podobu vektoru

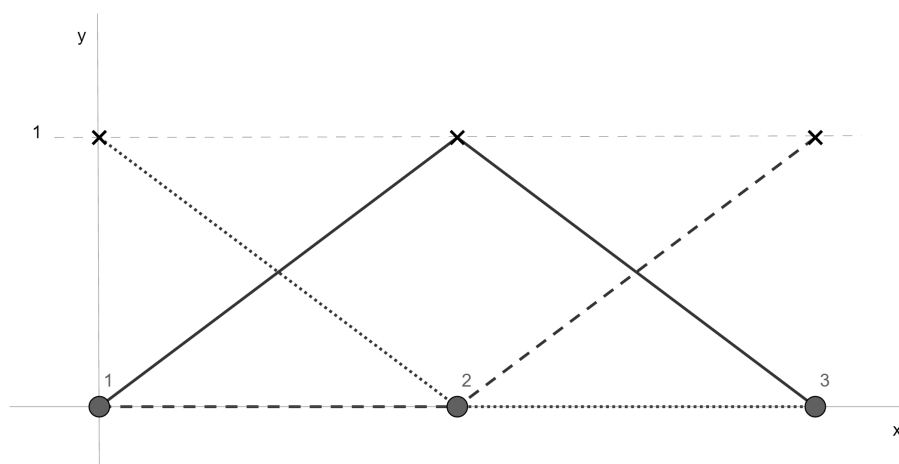
$$(0, \frac{1}{2}, 1, \frac{1}{2}, 0)^T$$

a pořadí sloupce v matici prolongace by odpovídalo číslu uzlu na jemné síti.

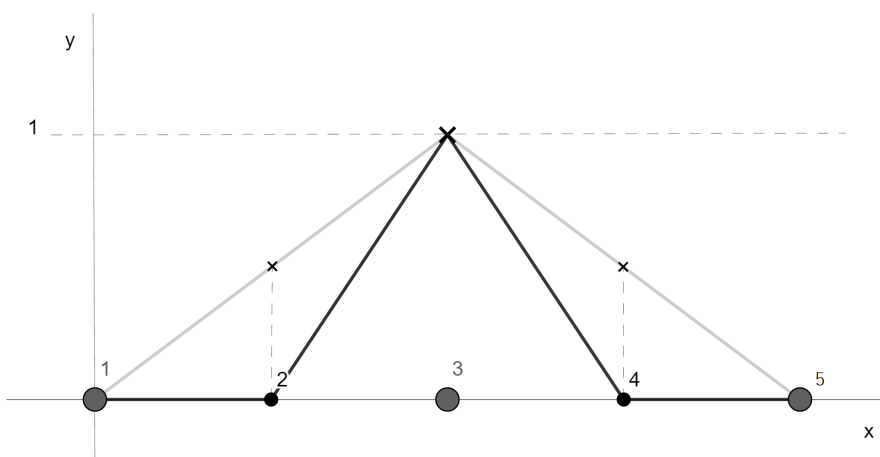
Nyní se pokusíme nastínit, jak bychom postupovali ve dvoudimensionálním případě. Na Obrázku 17 je vyobrazena hrubá síť a naznačena bazová funkce pro vnitřní uzel číslo 5, kdy 1 se na ose y nachází nad tímto bodem. Na Obrázku 18 je vyobrazena původní jemná síť. Pokud budeme postupovat odspoda nahoru a zleva doprava tak, jak jsou uzly očíslovány, pak třináctý sloupec bazové funkce pro vybraný uzel vypadá následovně

$$(0, 0, 0, 0, 0, 0, \frac{1}{4}, \frac{1}{2}, \frac{1}{4}, 0, 0, \frac{1}{2}, 1, \frac{1}{2}, 0, 0, \frac{1}{4}, \frac{1}{2}, \frac{1}{4}, 0, 0, 0, 0, 0, 0)^T.$$

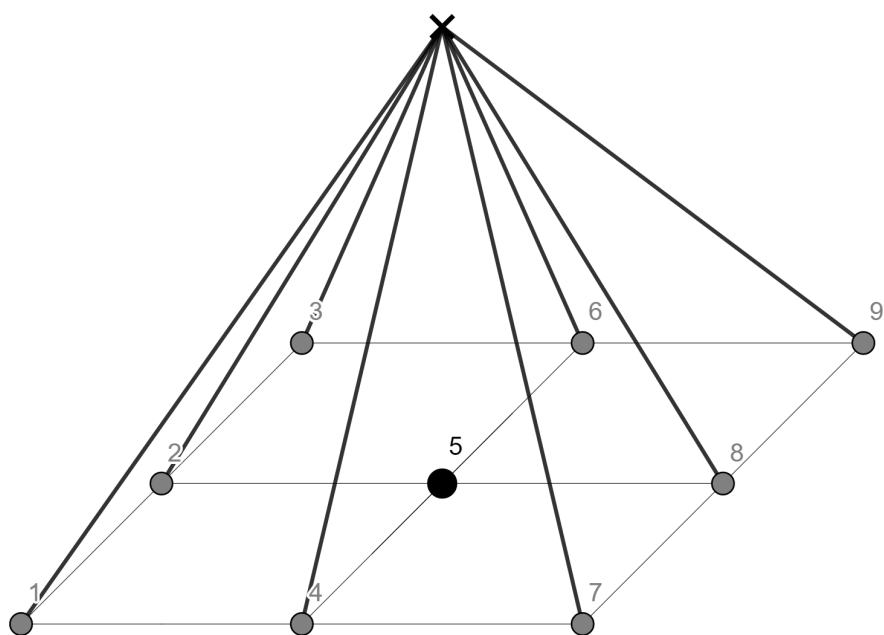
Pokud stejným způsobem sestavíme sloupec matice prolongace pro každý vnitřní uzel naší MKP soustavy, dosáhneme tak kýžené matice prolongace \mathbf{P} .



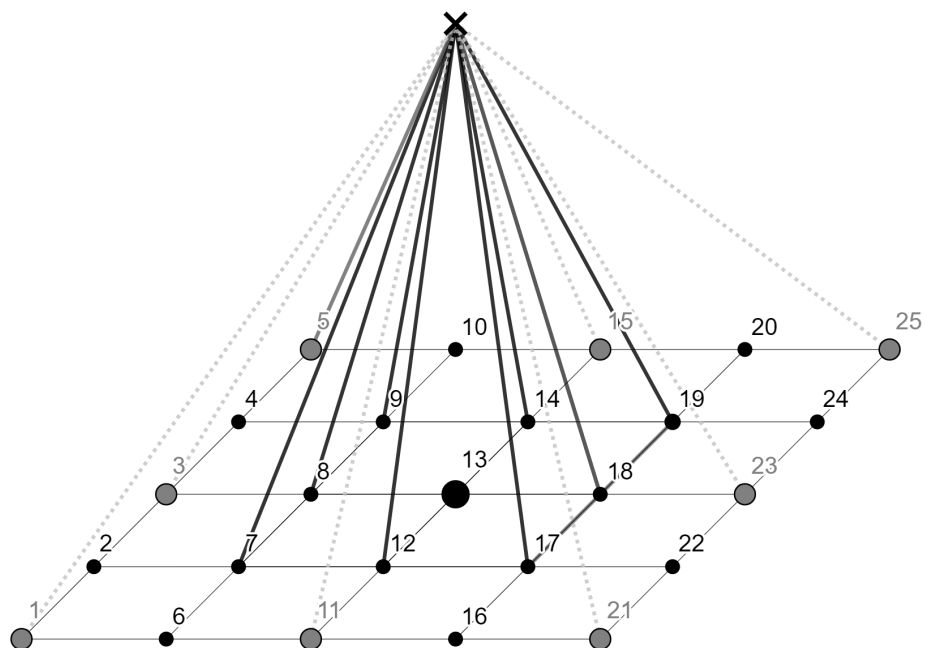
Obrázek 15: Ilustrace bázových funkcí hrubé sítě v jedné dimenzi; $\varphi_1(x)$ bázová funkce uzlu 1 značena tečkovanou čarou; $\varphi_2(x)$ bázová funkce uzlu 2 značena plnou čarou; $\varphi_3(x)$ bázová funkce uzlu 3 značena přerušovanou čarou



Obrázek 16: Ilustrace bázové funkce uzlu jemné sítě (černě) ve srovnání s bázovou funkcí původního uzlu 2 hrubé sítě (šedě) v jedné dimenzi



Obrázek 17: Ilustrace bázové funkce uzlu 5 hrubé sítě ve dvou dimenzích



Obrázek 18: Ilustrace bázové funkce uzlu jemné sítě (černě) ve srovnání s bázovou funkcí původního uzlu 5 hrubé sítě (šedě) ve dvou dimenzích

4.2 Vlastní implementace

Naznačme si nyní jak probíhala implementace metody dvou sítí.

Samotný algoritmus metody dvou sítí jsme popsali již v teoretickém úvodu této kapitoly, shrňme jej nyní do jednoduchého pseudoalgoritmu (Algoritmus 2):

Algoritmus 2 Metoda dvou sítí

Vstup: $n, r^h, A^h, itSmooth$

```
u = zeros(n, 1)
rh = Richardson(rh, A, u, itSmooth)      %Pre-Smoothing
AH = PT * Ah * P
rH = PT * r
eh = P * (AH \ rH)
u = u + eh
rh = rh - Ah * ej
rh = Richardson(rh, A, u, itSmooth)      %Post-Smoothing
```

Výstup: r^h

V teoretickém úvodu této kapitoly, jsme rovněž podrobně popsali sestavení matice \mathbf{P} . V případě takové globální matice \mathbf{A} , jakou jsme vytvořili v podkapitole o vlastní implementaci DDM, není číslování uzlů spořádáno tak příjemně, jako na například na Obrázku 18. Proto si při sestavení matice prolongace musíme dát pozor, abychom dodrželi takový systém číslování uzlů, jaký matice \mathbf{A} skutečně obsahuje. Nicméně algoritmus sestavení matice \mathbf{P} je analogický.

Metodu dvou sítí pak iterujeme tak dlouhou, dokud poměr norem nového rezidua r^h a původního rezidua soustavy r_0 ($r_0 = b^h - A^h u_0^h$) neodpovídá námi zvolené přesnosti.

5 Optimalizace metody rozložení oblasti pomocí metody dvou sítí ve 2D

Připomeňme si „tříkrokovou“ metodu z Kapitoly 3. Jak už jsme říkali, zde v prvním a třetím kroku řešíme MKP soustavu N rovnic pomocí Gaussovy eliminace. Ukažme si v následující podkapitole, proč bychom se této metodě rádi vyhnuli.

5.0.1 Gaussova eliminace

Algoritmus Gaussovy eliminace můžeme vidět níže (Algoritmus 3).

Protože platí

$$\sum_{k=1}^n k = \frac{n(n+1)}{2}, \quad \sum_{k=1}^2 k^2 = \frac{n(n+1)(2n+1)}{6},$$

kde n je počet řádků, respektive sloupců matice \mathbf{A} , dojdeme k závěru, že v přímém chodu GEM, kdy matici převádíme na trojúhelníkový tvar, potřebujeme k výpočtu multiplikátorů m_{ik}^{k-1}

$$\sum_{k=1}^{n-1} (n-k) = (n-1)n - \frac{(n-1)n}{2} = \frac{n(n-1)}{2}$$

dělení a k výpočtu prvků a_{ij}^k potřebujeme

$$\sum_{k=1}^{n-1} (n-k)(n-k+1) = \sum_{k=1}^{n-1} n(n+1) - k(2n+1) + k^2 = \frac{n^3 - n}{3}$$

násobení a stejně tolik sčítání. Při zpětném chodu GEM, tj. eliminaci proměnných, je třeba

$$\frac{n^2 + n}{2}$$

násobení a

$$\frac{n^2 - n}{2}$$

sčítání. Celkem v našem případě tedy potřebujeme $\frac{n^3}{3} + n^2 - \frac{n}{3}$ násobení a dělení a $\frac{n^3}{3} + \frac{n^2}{2} - \frac{5n}{6}$ sčítání. Z hlediska počtu binárních operací potřebných k realizaci algoritmu jsou operace násobení a dělení velmi obdobné. Protože výpočetní náročnost násobení je několikanásobně vyšší než ta, kterou vyžaduje operace sčítání, budeme o efektivnosti výpočtu rozhodovat v závislosti na počtu násobení. K realizaci GEM je tedy potřeba $\frac{n^3}{3} + \mathbf{O}(n^2)$ násobení. To znamená, že když v soustavě počet rovnic zdvojnásobíme, vzroste doba výpočtu osmkrát, a proto není GEM ideální metodou pro řešení MKP soustav.

Algoritmus 3 Gaussova eliminační metoda

Vstup: n, A, b

```
for  $k = 1, 2, \dots, n - 1$  do
  for  $i = k + 1, k + 2, \dots, n$  do
     $m_{ik}^{k-1} = -\frac{a_{ik}}{a_{kk}}$  (pokud  $a_{kk} \neq 0$ )
    for  $j = k + 1, k + 2, \dots, n + 1$  do
       $a_{ij} = a_{ij} + m_{ik}^{k-1} a_{kj}$ 
    end for
  end for
end for
for  $i = n, n - 1, \dots, 1$  do
   $u_i = \frac{1}{a_{ii}} \left( a_{i,n+1} - \sum_{j=i+1}^n a_{ij} u_j \right)$ 
end for
Výstup:  $u$ 
```

Odtud plyne hlavní myšlenka této práce. Výpočetní náročnost jedné iterace multigridu je lineární, a proto v „tříkrokové“ metodě budeme lineární soustavu N rovnic řešit pomocí metody dvou sítí. Nicméně protože GEM je přímý řešič zatím co metoda dvou sítí řešení aproximuje, můžeme předpokládat, že výsledný počet iterací předpokládané metody sdružených gradientů vzroste. Do jaké míry se tato skutečnost projeví, si ověříme při numerických testech.

5.1 Numerické experimenty

Nyní bychom rádi porovnali rychlost konvergence pro DDM využívající Gaussovu eliminaci a DDM optimalizované metodou dvou sítí.

5.1.1 Rychlost konvergence

Stejně jako v Kapitole 3, kde sledujeme rychlost konvergence DDM využívající GEM, i nyní provedme testy pro $\frac{H}{h} = 4$, $N = 2, 4, 6, \dots, 30$ a $\frac{H}{h} = 16$, $N = 2, 4, 6, \dots, 30$. Tentokrát sledujeme počet iterací předpokládané metody sdružených gradientů pro DDM, kde jsme GEM nahradili metodou dvou sítí. Výsledky jsou uvedeny v Tabulkách číslo 5 a 6.

Zajímá nás, zda jsme výpočetní náročnost nesnížili na úkor rychlosti konvergence. Pokud srovnáme Tabulky 3 a 5 můžeme vidět, že jsou výsledky identické. Po srovnání Tabulek číslo 4 a 6 zjišťujeme, že v případě DDM optimalizované metodou dvou sítí, trvá o něco déle, než se počet iterací ustálí, nicméně, i tak od určitého počtu podoblastí hrubé diskretizace, bude počet iterací nezávislý na této proměnné.

Vzhledem k výpočetnímu času, který výrazněji ušetříme spíše na větších soustavách, jsou výsledky příznivé. Nesmíme však zapomenout, že jsme multigrid nahradili metodou dvou sítí, která při výpočtu chyby na hrubé síti využívá GEM. V tuto chvíli se může zdát, že jsme se výpo-

četně náročné GEM vlastně nevyhnuli, proto připomeňme, co jsme zmínili na začátku kapitoly o multigridu. Smyslem multigridu je rekurzivně vytvářet hrubší síť tak dlouho, dokud nebude výpočetní náročnost určení chyby hrubé sítě pomocí GEM zanedbatelná. Cílem této práce je ukázat, že k určení DDM předpodmínění, můžeme GEM v „tříkrokové“ metodě nahradit některou z multigridních metod, což se nám také povedlo. Pokud bychom chtěli na superpočítači testovat větší soustavy, bylo by nutné namísto metody dvou sítí využít víceúrovňový multigrid. Dalším faktem, který nesmíme zanedbat je, že víceúrovňový multigrid bude poskytovat méně přesnou aproximaci, než metoda dvou sítí. Může se proto stát, že rychlost konvergence se - v případě využití víceúrovňového multigridu - pro nižší počet podoblastí hrubé diskretizace nebude tolik podobat rychlosti konvergence DDM využívající GEM. To můžeme do určité míry regulovat rozumnou volbou tolerance v iteračním cyklu, kde srovnáváme normy původního a nového residua, tak jak jsme popsali v podkapitole o vlastní implementaci metody dvou sítí. Nicméně ve vztahu s rychlostí konvergence víceúrovňového multigridu je pro nás stěžejní kombinace dvou poznatků této práce. Prvním z nich je, že rychlost konvergence předpodmíněné metody sdružených gradientů je přímo-úměrná číslu podmíněnosti matice řešené soustavy. Druhým poznatkem jsou výsledky numerických testů při zkoumání čísla podmíněnosti. Tehdy jsme si ověřili, že od určitého počtu podoblastí hrubé diskretizace se číslo podmíněnosti přestane měnit. Kombinací těchto informací můžeme vytvořit dobře podložený předpoklad, že se i v případě horší aproximace víceúrovňového multigridu počet iterací předpodmíněné metody sdružených gradientů relativně brzo ustálí.

$\frac{H}{h} = 4, N =$	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Počet iterací	5	5	10	9	11	12	12	12	11	11	11	11	11	10	10
$\frac{H}{h} = 4, N =$	17	18	19	20	21	22	23	24	25	26	27	28	29	30	
Počet iterací	10	10	10	10	10	10	10	10	10	10	10	10	10	10	

Tabulka 5: Počet iterací předpodmíněné metody sdružených gradientů za využití metody dvou sítí pro $\frac{H}{h} = 4$ a měnící se parametr N

$\frac{H}{h} = 16, N =$	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Počet iterací	6	6	12	11	14	13	14	13	14	13	13	13	13	13	13
$\frac{H}{h} = 16, N =$	17	18	19	20	21	22	23	24	25	26	27	28	29	30	
Počet iterací	13	13	13	13	13	13	13	13	13	13	13	13	13	13	13

Tabulka 6: Počet iterací předpodmíněné metody sdružených gradientů za využití metody dvou sítí pro $\frac{H}{h} = 16$ a měnící se parametr N

6 Závěr

Cílem této práce bylo vytvoření nástroje k optimalizaci výpočetní náročnosti při řešení velkých MKP soustav pomocí paralelizace. Konkrétně jsme se pak měli zaměřit na nahrazení Gaussovy eliminační metody (GEM) s kvadratickou složitostí multigridem, jehož složitost je pouze lineární. Vytvořený algoritmus jsme měli testovat na dvoudimenzionální úloze vedení tepla, tedy diskretizované úloze eliptické parciální diferenciální rovnice (PDR) druhého řádu.

V druhé kapitole jsme si popsali řešení úloh eliptických PDR druhého řádu pomocí MKP. Postupně jsme se věnovali podobě klasického řešení PDR, definovali jsme slabé řešení úlohy, popsali algoritmus tvorby MKP soustavy $Au = b$ a zmínili některé důležité vlastnosti matice tuhosti \mathbf{A} .

Ve třetí kapitole jsme se věnovali metodě rozložení oblasti (DDM), která nám umožňuje optimalizovat výpočetní náročnost při hledání řešení u MKP soustavy $Au = b$. Cílem této metody je najít paralelní SPD předpokládání $\hat{\mathbf{A}}^{-1}$ tak, abychom získali lépe podmíněnou MKP soustavu a mohli ji efektivně řešit předpokládáním metodou sdružených gradientů.

V teoretické části této kapitoly jsme rozebrali způsob rozložení původní oblasti na podoblasti hrubé a jemné diskretizace, přeuspořádání bazových funkcí matice \mathbf{A} v závislosti na novém způsobu diskretizace a na závěr teoretické části jsme se propracovali k popisu způsobu sestavení paralelního SPD předpokládání $\hat{\mathbf{A}}^{-1}$. A ukázali si, že právě při tvorbě předpokládání využívá DDM výpočetně náročnou GEM.

Další částí třetí kapitoly je popis vlastní implementace metody rozložení oblasti, kde jsme podrobně rozebrali sestavení a podobu globální matice soustavy.

Na závěr kapitoly jsme prováděli numerické experimenty a ověřili si tak vlastnosti MKP soustavy optimalizované pomocí DDM. Nejprve jsme zkoumali číslo podmíněnosti

$$\kappa(\hat{\mathbf{S}}^{-1}\mathbf{S}) \approx \kappa(\hat{\mathbf{A}}^{-1}\mathbf{A}),$$

kde S je Schurův doplněk a \hat{S}^{-1} jeho modifikace. Těmito testy jsme získali cennou informaci, která říká, že od určitého počtu podoblastí hrubé diskretizace se hodnota čísla podmíněnosti ustálí nezávisle na navyšujícím se počtu podoblastí hrubé diskretizace. Dále jsme ověřili, že takto podmíněná soustava je i pro vyšší počet podoblastí jemné diskretizace stabilní. Stabilita soustavy pro vysoký počet podoblastí hrubé diskretizace pak vyplývá už z testování změn čísla podmíněnosti v závislosti na této proměnné. Při dalších testech jsme hledali řešení optimalizované MKP soustavy pomocí předpokládání metody sdružených gradientů. Jednou ze stěžejních informací při vytváření závěrů na základě numerických testů je, že rychlost konvergence metody sdružených gradientů je přímo úměrná číslu podmíněnosti matice soustavy, a my tak právě zde aplikujeme DDM předpokládání. Myšlenku algoritmu předpokládání metody sdružených gradientů jsme si v krátkosti popsali. V testech jsme sledovali rychlost konvergence při hledání řešení u pro zvyšující se počet podoblastí hrubé diskretizace. Ve výsledcích jsme se pak

opět vrátili k prvotnímu poznatku z numerických experimentů této práce. Vzhledem k tomu, že číslo podmíněnosti je od určitého počtu podoblastí hrubé diskretizace ustáleno, počet iterací předpokládané metody sdružených gradientů se v daném okamžiku přestane měnit nezávisle na zvyšujícím se počtu podoblastí hrubé diskretizace. Numerické experimenty ve třetí kapitole tedy potvrdili, že jsme díky DDM příjemným způsobem optimalizovali naši MKP soustavu.

V kapitole čtvrté jsme úvodní slovo věnovali multigridu, který je nejrychlejším dnes známým řešičem MKP soustav. Víceúrovňový multigrid jsme nechali pro tuto kapitolu stranou a pro účely aplikace jsme se zaměřili na metodu dvou sítí. Kapitola obsahuje podrobný algoritmus metody dvou sítí i sestavení matice prolongace, která je její nedílnou součástí. Na závěr kapitoly jsme se krátce věnovali vlastní implementaci metody dvou sítí.

Poslední kapitola se zabývá cílem této práce, tedy optimalizací metody rozložení oblasti pomocí multigridu. Zde je konečně ukázán důvod kvadratické složitosti GEM. Dále jsme provedli numerické experimenty srovnávající klasickou DDM a DDM optimalizovanou pomocí metody dvou sítí. V této kapitole jsme si ověřili, zda nahrazením přímého řešiče (GEM) aproximační metodou dvou sítí nesnižujeme výpočetní náročnost na úkor rychlosti konvergence.

Numerické experimenty obsažené v této kapitole ukázaly, že v případě optimalizované DDM se počet iterací předpokládané metody sdružených gradientů pro zvyšující se počet podoblastí hrubé diskretizace ustálí později. Nicméně opět se vracíme k prvotním výsledkům experimentů z třetí kapitoly. Číslo podmíněnosti soustavy se pro zvyšující se počet podoblastí hrubé diskretizace přestane od určitého okamžiku měnit. My tak budeme od konkrétní chvíle provádět stále stejný počet iterací předpokládané metody sdružených gradientů, nehledě zvyšující se počet podoblastí hrubé diskretizace. Došli jsme tedy ke stejnému závěru jako v případě DDM využívající GEM. Povedlo se nám tak v metodě rozložení oblasti nahradit GEM s kvadratickou složitostí výpočetně méně náročnou metodou dvou sítí.

V závěru poslední kapitoly se pak věnujeme možným důsledkům při nahrazení metody dvou sítí víceúrovňovým multigridem. V praxi bychom optimalizovanou DDM rádi aplikovali při řešení velkých soustav. Metoda dvou sítí při výpočtu aproximace chyby využívá GEM, a může se tak zdát, že jsme se této výpočetně náročné metodě nakonec nevyvarovali. Nicméně, jak bylo zmíněno v úvodu Kapitoly 4, myšlenkou víceúrovňového multigridu je rekurzivní vytváření hrubých sítí do té doby, než bude výpočetní náročnost aproximace chyby pomocí GEM zanedbatelná. Dalším rozdílem těchto dvou metod je, že metoda dvou sítí aproximuje výsledek mnohem rychleji než víceúrovňový multigrid. Může se tak stát, že pro menší počet podoblastí hrubé diskretizace nebude rychlost konvergence optimalizované DDM tolik podobná rychlosti konvergence DDM klasické jako v případě numerických experimentů této kapitoly. Zde si opět připomeňme numerické experimenty z Kapitoly 3. Od určitého počtu podoblastí hrubé diskretizace je číslo podmíněnosti matice soustavy ustáleno a pro zvyšující se počet podoblastí hrubé diskretizace je neměnné. My tak opět budeme od nějakého okamžiku (ač v případě multigridu o něco poz-

ději), provádět stále stejný počet iterací předpodmíněné metody sdružených gradientů nezávisle na této proměnné.

Tato práce poskytuje nástroj k optimalizaci výpočetní náročnosti při řešení velkých MKP soustav pomocí paralelizace. Kdy jsme metodu rozložení oblasti, která ve své klasické verzi obsahuje výpočetně náročnou Gaussovu eliminační metodu, optimalizovali nahrazením GEM „levnější“ metodou dvou sítí. Ukázali jsme, že v případě řešení velkých soustav rozložených na velký počet podoblastí hrubého dělení ve smyslu DDM, počet iterací předpodmíněné metody sdružených gradientů nezávisí na počtu těchto podoblastí. Dále jsme na základě výsledků z numerických experimentů ukázali, že můžeme metodu dvou sítí bez obav nahradit víceúrovňovým multigridem s lineární složitostí.

Na tuto práci by jistě bylo vhodné navázat praktickou aplikací paralelizovaného hledání řešení velké soustavy na superpočítači a podpořit závěry práce výsledky pro soustavy, které není možné řešit na běžném hardware.

Literatura

- [1] LUKÁŠ, BOUCHALA, VODSTRČIL, MALÝ. *2-dimensional primal domain decomposition theory in detail*. Applications of Mathematics 60, pp. 265-283, 2015
- [2] TOMŠÍK, J. *Numerické metody pro řídké matice*. Brno, 2014. 104 s. Diplomová práce na Přírodovědecké fakultě Masarykovy univerzity na ústavu matematiky a statistiky. Vedoucí diplomové práce Mgr. Jiří Zelinka, Dr.
- [3] MOŠOVÁ, V. *Numerické metody*. 1. vyd. Olomouc: Univerzita Palackého, 2003. 147 s. ISBN 80-244-0620-9.
- [4] HABALA, P. *Skriptum k předmětu Differential equations and numerical methods*. Praha: ČVUT fakulta elektrotechnická, Katedra matematiky, [cit. 2019-20-04].
- [5] SOBOTÍKOVÁ, V. *Metoda konečných prvků*. Matematická kolokvia katedry matematiky FEL ČVUT v Praze, 2018
- [6] BOUCHALA, J. *Variační metody*. Vysoká škola báňská – Technická univerzita Ostrava a Západočeská univerzita v Plzni, 2012.